Evropská unie
Evropský sociální fond
Operační program Zaměstnanost

# TECHNICKÁ DOKUMENTACE NÁSTROJE CZ-DRG PRE-GROUPER

Metodický materiál projektu DRG Restart

**Zpracoval autorský kolektiv pod vedením:** T. Pavlík, M. Bartůňková, P. Klika, J. Linda, L. Dušek

**Autoři:** P. Klika, K. Kupčák, T. Pavlík, Z. Bortlíček, M. Uher, J. Klika

Verze: CZ-DRG 2027

Verze dokumentu: 2

Datum: 3. 9. 2026

# Obsah

Historie verzí dokumentu ... 5
Úvod ... 10
Účel nástroje ... 10
Změna SW prostředí – Java 17 ... 10
~~Nástroj pro tvorbu procesů datové integrace~~ ... 10
Struktura technické dokumentace ... 11
Testování funkčnosti nástroje pre-grouper a jeho výstupů ... 11
Základní architektura nástroje pre-grouperu ... 12
Příprava dat ve formátu CSV ... 12
Seřazení vstupních souborů ... 12
Příprava metadat pro random-access-file (RAF) ... 12
Paralelní zpracování datových dávek ... 13
Proces agregace dat do hospitalizačních případů ... 13
Porovnání výstupu s Talend verzí ... 13
Import projektu pre-grouper v nástroji NetBeans ... 14
Instalace nástroje NetBeans ... 14
Otevření projektu pro pre-grouper CZ-DRG ... 15
Popis automaticky generované dokumentace procesu pre-grouper ... 18
~~Základní členění procesu~~ ... 18
~~Procesní diagramy~~ ... 18
~~Datové diagramy~~ ... 19
~~Ukázka popisu komponenty v generované dokumentaci~~ ... 19
~~Import projektu pre-grouper v nástroji datové integrace~~ ... 22
~~Instalace nástroje Talend~~ ... 22
~~Import projektu pre-grouperu~~ ... 23
~~Dokumentace nástroje Talend~~ ... 25
Popis jednotlivých částí procesu pre-grouper ... 26
pgp_service_master ... 26
pgp_service_main ... 27
pgp_master ... 28
pgp_db_input ... 29
pgp_validace_master ... 31
pgp_validace_k0_paralelni ... 31
pgp_doklady02_validace_k0 ... 32

pgp_dokladyOstatni_validace_k0 ... 33
pgp_polozky_validace_k0 ... 34
pgp_doklady02_validace_k1 ... 35
pgp_validace_info ... 36
pgp_inicialni_paralelni ... 37
pgp_dgvedlejsi_deduplikace ... 39
pgp_vdg_zavaznosti ... 40
pgp_01_hosp_pripady ... 41
pgp_02_doklady ... 43
pgp_03_polozky ... 44
pgp_validace_pripadu ... 45
pgp_04a_doklady_pripady_master ... 46
pgp_doklady02Pripady_master ... 47
pgp_doklady02Pripady_hlavni ... 48
kontrola_napojeni_na_vice_pripadu_fce ... 50
rozdeleni_dle_odbornosti_ARLP_fce ... 51
pgp_doklady06Pripady_master ... 52
pgp_doklady06Pripady_hlavni ... 53
pgp_doklady06Pripady_extramuralni ... 55
pgp_doklady06Pripady_intramuralni ... 57
pgp_doklady06Pripady_nejasnePriraditelna ... 58
spravneNapojene_unite ... 59
pgp_dokladyPripady_vystup ... 60
pgp_04b_polozky_ciselniky ... 62
pgp_05a_agregace_polozek ... 65
pgp_05b_agregace_dalsi ... 66
pgp_05c_sestaveni_vv ... 69
pgp_clear_tempdir ... 70
pgp_db_output ... 71
pgp_service_kontrola_dbdata ... 72
pgp_temp_db_output ... 73
~~Příloha 1: Testování výstupů pre-grouperu~~ ... 75
~~Přiřazení dokladů 02 k hospitalizačním případům~~ ... 75
~~Přiřazení dokladů 06 k hospitalizačním případům~~ ... 75
~~Sestavení hospitalizačního případu~~ ... 76

~~Náplň a vlastnosti hospitalizačního případu~~ ..... 76

# Historie verzí dokumentu

| Verze/Revize | Datum | Popis změn |
|---|---|---|
| 1/1 | 14. 05. 2018 | První oficiální publikovaná verze technické dokumentace nástroje CZ-DRG Pre-Grouper |
| 2/2 | 25. 10. 2019 | pgp_master<br>• nová verze procesu 1.1<br>pgp_validace_master<br>• nová verze procesu 1.1<br>pgp_doklady02_validace_k1<br>• nová verze procesu 1.5<br>pgp_validace_info<br>• nově přidaný proces (verze 0.1)<br>pgp_inicialni_paralelni<br>• nová verze procesu 1.1<br>pgp_01_hosp_pripady<br>• nová verze procesu 1.5<br>pgp_02_doklady<br>• nová verze procesu 1.2<br>pgp_03_polozky<br>• nová verze procesu 1.1<br>pgp_validace_pripadu<br>• nově přidaný proces (verze 0.1)<br>pgp_04a_doklady_pripady_master<br>• nová verze procesu 1.1<br>pgp_doklady02Pripady_hlavni<br>• nová verze procesu 1.1<br>spravneNapojene_unite<br>• nová verze procesu 1.1<br>pgp_04b_polozky_ciselniky<br>• nová verze procesu 1.2<br>pgp_05a_agregace_polozek<br>• nová verze procesu 1.3<br>pgp_05b_agregace_dalsi<br>• nová verze procesu 1.7<br>pgp_05c_sestaveni_vv<br>• nová verze procesu 1.1<br>pgp_clear_tempdir<br>• nová verze procesu 1.1<br>pgp_temp_db_output<br>nově přidaný proces (verze 0.3) |
| 3/0 | 28. 2. 2020 | pgp_04b_polozky_ciselniky<br>• nová verze procesu 1.3<br>pgp_doklady06Pripady_master<br>• nová verze procesu 1.1<br>pgp_05b_agregace_dalsi<br>• nová verze procesu 1.8<br>pgp_05c_sestaveni_vv |

| | | |
|---|---|---|
| | | • nová verze procesu 1.2<br>pgp_db_output<br>• nová verze procesu 1.1 |
| 3/1 | 10. 9. 2020 | pgp_inicialni_paralelni<br>• nová verze procesu 1.2<br>pgp_dgvedlejsi_deduplikace<br>• nově přidaný proces (verze 0.1)<br>pgp_01_hosp_pripady<br>• nová verze procesu 1.6<br>pgp_db_input<br>• nová verze procesu 1.1<br>pgp_master<br>• nová verze procesu 1.2 |
| 3/1 | 17.12.2020 | pgp_01_hosp_pripady<br>• nová verze procesu 1.7 (oprava chyby při vytváření HP)<br>pgp_05a_agregace_polozek<br>• nová verze procesu 1.4 (oprava chyby – přidání filtru pro ZUP pouze v období hospitalizačního případu HP) |
| 4/0 | 30.12.2020 | pgp_05b_agregace_dalsi<br>• nová verze procesu 1.9 |
| 4/1 | 15.7.2021 | Použití Talend verze 7.3.1 místo verze 6.4.1<br>code/routines/http_service/HttpHandlerService<br>• přidání code routines pro service procesy<br>code/routines/http_service/ HttpQuequeData<br>• přidání code routines pro service procesy<br>pgp_service_master<br>• nový hlavní proces pregrouperu 0.1<br>pgp_service_main<br>• nový hlavní proces aplikace v režimu služby 0.2<br>pgp_service_kontrola_dbdata<br>• nový proces spouštěný v režimu služby 0.1<br>pgp_master<br>• nová verze procesu 1.5<br>pgp_doklady06pripady_hlavni<br>• nová verze procesu 1.1<br>pgp_doklady06pripady_intramuralni<br>• nová verze procesu 1.1<br>pgp_db_input<br>• nová verze procesu 1.2<br>spravnenapojene_unite<br>• nová verze procesu 1.2<br>pgp_05c_sestaveni_vv<br>• nová verze procesu 1.3<br>pgp_temp_db_output<br>• nová verze procesu 0.4<br>pgp_db_output<br>• nová verze procesu 1.2<br>pgp_05b_agregace_dalsi<br>• nová verze procesu 1.10<br>pgp_doklady06pripady_nejasnepriraditelna |

| | | |
|---|---|---|
| | | • nová verze procesu 1.1<br>pgp_doklady06pripady_extramuralni<br>• nová verze procesu 1.1 |
| 5/0 | 10.12.2021 | pgp_04b_polozky_ciselniky<br>• nová verze procesu 1.4<br>pgp_05a_agregace_polozek<br>• nová verze procesu 1.5<br>pgp_05b_agregace_dalsi<br>• nová verze procesu 1.11 |
| 5/0 | 22.2.2022 | pgp_service_kontrola_dbdata<br>• nová verze procesu 0.2 |
| 5/1 | 30.6.2022 | pgp_04b_polozky_ciselniky<br>• nová verze procesu 1.5<br>pgp_05a_agregace_polozek<br>• nová verze procesu 1.6<br>pgp_service_kontrola_dbdata<br>• nová verze procesu 0.3<br>pgp_master<br>• nová verze procesu 1.6 |
| 5/1 | 9.9.2022 | pgp_db_input<br>• nová verze procesu 1.3<br>pgp_db_output<br>• nová verze procesu 1.3<br>pgp_temp_db_output<br>• nová verze procesu 0.5<br>pgp_service_kontrola_dbdata<br>• nová verze procesu 0.4 |
| 5/1 | 30.11.2022 | pgp_04b_polozky_ciselniky<br>• nová verze procesu 1.6<br>pgp_05a_agregace_polozek<br>• nová verze procesu 1.9<br>pgp_05b_agregace_dalsi<br>• nová verze procesu 1.13<br>pgp_doklady02_validace_k1<br>• nová verze procesu 1.6<br>pgp_master<br>• nová verze procesu 1.7 |
| 6/0 | 28.2.2023 | pgp_db_output<br>• nová verze procesu 1.4<br>pgp_04b_polozky_ciselniky<br>• nová verze procesu 2.0<br>pgp_05b_agregace_dalsi<br>• nová verze procesu 2.0<br>pgp_clear_tempdir<br>• nová verze procesu 2.0<br>pgp_master<br>• nová verze procesu 2.0 |
| 6/0 | 1.9.2023 | vstupni_veta_0.4<br>• úprava formátu položky IDZZ |

| | | |
|---|---|---|
| | | pgp_db_output<br>• nová verze procesu 2.0<br>pgp_05a_agregace_polozek<br>• nová verze procesu 2.0<br>pgp_05c_sestaveni_vv<br>• nová verze procesu 2.0 |
| 7/0 | 1.3.2024 | pgp_dgvedlejsi_deduplikace<br>• nová verze procesu 0.2<br>pgp_vdg_zavaznosti<br>• nově přidaný proces (verze 0.1)<br>pgp_01_hosp_pripady<br>• nová verze procesu 2.0<br>pgp_02_doklady<br>• nová verze procesu 1.3<br>pgp_05b_agregace_dalsi<br>• nová verze procesu 2.1 |
| 7/0 | 31.8.2024 | pgp_02_doklady<br>• nová verze procesu 1.4<br>pgp_05b_agregace_dalsi<br>• nová verze procesu 2.2 |
| 2026 | 31.5.2025 | pgp_master<br>• změna verze v map na "CZ-DRG Pregrouper 2026 - 2026r0-250531"<br>pgp_doklady06pripady_hlavni<br>• přidány sloupce DGV a DGVT na vstupu<br>**PGP_DOKLADYPRIPADY_VYSTUP**<br>• přidány sloupce DGV a DGVT<br>**PGP_DOKLADY02_VALIDACE_K0**<br>• přidány sloupce DGV a DGVT<br>pgp_doklady02_validace_k1<br>• přidány sloupce DGV a DGVT<br>pgp_01_hosp_pripady<br>• přidány sloupce DGV a DGVT<br>pgp_02_doklady<br>• přidány sloupce DGV a DGVT na vstupu<br>pgp_04b_polozky_ciselniky<br>• přidány sloupce DGV a DGVT<br>• přidáno načtení nového čísleníku polytrauma_dny<br>• grafické přeskládání některých tasků načítajících číselníky<br>• do výstupu pgp_04_temp přidán sloupec "je_v_cis_polytrauma_dny" s logickými hodnotami 0 nebo 1 stejně jako u ostatních připojených číselníků<br>pgp_05a_agregace_polozek<br>• přidány sloupce DGV a DGVT<br>pgp_05b_agregace_dalsi<br>• přidány sloupce DGV a DGVT<br>• přidána nový větev pro zpracování polytrauma dnů<br>• na základě sloupce "je_v_cis_polytrauma_dny" vypočte počet polytrauma dnů |

| | | |
|---|---|---|
| | | • vložení výsledného počtu polytrauma dnů do sloupce "KP3"<br>• změna verze v map_agregace_ciselniky na "2026"<br>pgp_05c_sestaveni_vv<br>• přidány sloupce DGV a DGVT<br>• ve sloupci "KP3" se nyní propisuje počet polytrauma dnů (prakticky se ale nic nemění pro KP3)<br>pgp_db_input<br>• přidány sloupce DGV a DGVT<br>pgp_db_output<br>• přidány sloupce DGV a DGVT<br>• změna z NULL pro KP3 na propsání hodnoty z předchozí části<br>pgp_dgvedlejsi_deduplikace<br>• přidány sloupce DGV a DGVT<br>pgp_vdg_zavaznosti<br>• přidány sloupce DGV a DGVT<br>• úprava .java skriptu pro zpracování nových DGVT sloupců<br>pgp_validace_pripadu<br>• přidány sloupce DGV a DGVT |
| 2027 | 15.4.2026 | Přepis technické dokumentace pro novou implementaci v jazyku JAVA<br>• Doplněn popis základní architektury procesu<br>• Doplněn postup importu do vývojového nástroje<br>• Odstranění obrázků a odkazů na původní automatickou dokumentaci TALEND<br>• Odstranění již neaktuálních částí |

# Úvod

## Účel nástroje

Program pre-grouper slouží k vytvoření podkladů pro klasifikační algoritmus systému CZ-DRG, pro vytvoření takzvaných vstupních datových vět grouperu (rozhraní je popsané v samostatném dokumentu DATOVÉ ROZHRANÍ NÁSTROJE CZ-DRG GROUPER). Vstupem pre-grouperu jsou soubory nebo databázové tabulky (resp. pohledy) odpovídající datovému rozhraní pre-grouperu, které je popsáno v samostatné příloze DATOVÉ ROZHRANÍ NÁSTROJE CZ-DRG PRE-GROUPER. Výstupem je soubor nebo databázová tabulka obsahující sestavené hospitalizační případy v podobě vstupní datové věty grouperu.

Pre-grouper implementuje postup popsaný v dokumentu METODIKA SESTAVENÍ HOSPITALIZAČNÍHO PŘÍPADU V SYSTÉMU CZ-DRG. Tento postup je závazný pro použití klasifikačního systému CZ-DRG. Případné vlastní implementace pre-grouperu musí vést ke stejným výsledkům jako tento nástroj.

## Změna SW prostředí – Java 17

Nástroj pre-grouper je CLI (řádková) aplikace implementovaná od verze 2027 přímo v jazyku Java 17 (na rozdíl od předchozích verzí, které byly vytvořeny s využitím integračního systému Talend Open Studio for Data Integration). **Důvodem přechodu od tohoto nástroje je ukončení vývoje a podpory tohoto původního nástroje a nahrazení jeho placenou verzí v jiné technologii. Zároveň došlo k povýšení minimální podporované verze Java na verzi 17 z důvodu, že jde o nejnižší verzi, která má od roku 2027 zajištěnu podporu včetně oprav kritických chyb a zabezpečení (tzv. Long-time support, LTS).**

Pre-grouper ~~je~~ i nadále zůstává multiplatformní, přenositelný, snadno spravovatelný a bezpečný (za předpokladu pravidelné aktualizace systému, JRE a vlastního nástroje pre-grouper). Postup instalace a použití nástroje je popsán v dokumentu UŽIVATELSKÝ MANUÁL NÁSTROJE CZ-DRG PRE-GROUPER.

Původní verze procesu agregace dokladů a položek do hospitalizačních případů (Pregrouper) byla implementovaná v externím nástroji Talend. Vzhledem k tomu, že nová verze v Javě vychází přímo z původní předlohy, je možné využívat původní Talend dokumentaci pro většinu kroků v procesu, třídy programu jsou pro jednodušší orientaci v kódu pojmenovány podle názvů v Talend dokumentaci.

Některé kroky procesu mají drobné změny – tam kde nedávalo smysl ponechávat proces naprosto identický, a některé kroky jsou navíc.

~~Nástroj pro tvorbu procesů datové integrace~~

~~Program pre-grouper je implementován v jazyku Java. Nevznikl však klasickým programováním celé aplikace, ale byl navržen ve specializovaném nástroji pro tvorbu procesů datové integrace – **Talend Open Studio for Data Integration**. Tento program umožňuje vytváření transformačních datových procesů v grafickém uživatelském rozhraní (GUI) v podobě napojování jednotlivých komponent do schémat datových či procesních toků. Na základě takto navržených procesů dochází k automatickému generování kódu v jazyce Java. Výsledný proces lze následně zkompilovat a spustit přes příkazovou řádku na jakémkoliv zařízení s instalovaným prostředím pro spouštění Java aplikací.~~

~~Výhodu použití tohoto přístupu oproti klasickému programování je vizuální a dokumentační element, který umožnuje snazší pochopení vytvořených procesů i netechnicky zaměřeným uživatelům.~~

## Struktura technické dokumentace

~~První kapitola vysvětluje, jak pracovat s dokumentací nástroje Talend a jaká je struktura celého procesu – viz **POPIS AUTOMATICKY GENEROVANÉ DOKUMENTACE PROCESU PRE-GROUPER**. Na jednom příkladu jsou ukázány jednotlivé části dokumentace a způsob popisu částí procesů pre-grouperu.~~

~~V druhé části technické dokumentace je popsán postup pro nainstalování nástroje Talend Open Studio for Data Integration a nahrání (import) implementovaných procesů programu pre-grouper do nástroje – **IMPORT PROJEKTU PRE-GROUPER V NÁSTROJI DATOVÉ INTEGRACE**. Uživatel programu si tak může jednotlivé procesy zobrazit a vyzkoušet upravit či spustit přímo v prostředí daného nástroje.~~

První část technické dokumentace **ARCHITEKTURA APLIKACE PRE-GROUPERU** popisuje základní architekturu nástroje – principy programových tříd a jejich vlastností a použití v procesu.

V druhé části je popsán postup pro nainstalování vývojového nástroje NetBeans a nahrání (import) projektu programu pre-grouper do něj – **IMPORT PROJEKTU PRE-GROUPER V NÁSTROJI NETBEANS**. Uživatel programu si tak může zobrazit zdrojový kód programu a vyzkoušet upravit či spustit přímo v prostředí tohoto nástroje.

Poslední část technické dokumentace obsahuje popis jednotlivých procesů programu pre-grouper – **POPIS JEDNOTLIVÝCH ČÁSTÍ PROCESU PRE-GROUPER**. Popis procesů se řídí danou šablonou. Nejprve je uvedena úvodní tabulka ***Základní popis procesu***, kde jsou shrnuty základní údaje o procesu – ~~odkaz na automaticky generovanou technickou dokumentaci,~~ související procesy (nadřazený job, předchozí a následující job) a seznam vstupních a výstupních dat do/z procesu. Následně je připojen krátký popis funkčnosti a významu procesu~~ doplněný o náhledový diagram z GUI prostředí nástroje Talend~~. U některých klíčových procesů jsou doplněny i další obrázky a parametry. U vybraných procesů je na konci připojena navíc tabulka ***Validační soubory vytvořené v rámci procesu*** uvádějící seznam souborů uložených ve složce *<tempdir>*, tyto soubory obsahují klíčové údaje pro případné další validace fungování procesu pre-grouperu. Jsou zde například seznamy dokladů, které byly z důvodu nalezené chyby vyřazeny ze zpracování nebo seznamy dokladů, které nebyly z konkrétního důvodu přiřazeny k žádnému hospitalizačnímu dokladu.

## Testování funkčnosti nástroje pre-grouper a jeho výstupů

Nástroj pre-grouper byl testován nad řadou testovacích dat, které ověřovaly jeho správnou funkcionalitu. ~~Za tímto účelem byla připravena sada testů, která ověřovala správnost výsledných sestavených hospitalizačních případů ze vstupních dat. Přílohou dokumentace je kompletní seznam testů, které takto byly implementovány pro testování výstupů z pre-grouperu za účelem zajištění jejich korektnosti – **PŘÍLOHA 1: TESTOVÁNÍ VÝSTUPŮ PRE-GROUPERU**.~~

Dále bylo provedeno uživatelské testování funkčnosti aplikace pre-grouperu pro práci nad různými operačními systémy, databázovými servery a vstupními daty.~~ Testování rychlosti s různým nastavením dostupných systémových prostředků nad vzorovými daty je dokumentováno v **UŽIVATELSKÉM MANUÁLU NÁSTROJE PRE-GROUPER KLASIFIKAČNÍHO SYSTÉMU CZ-DRG.**~~

# Základní architektura nástroje pre-grouperu

Následující kapitola popisuje základní členění tříd a princip fungování programu pre-grouper. Program má možnosti vstupu/výstupu souborů typu csv anebo databázových tabulek pro jednorázový větší balík dat v rámci jednoho běhu anebo může být spuštěný jako služba, kde přijímá filtr dat v databázi podle položky RUN_ID anebo ID_DOKLADU v http requestech. Všechny tyto možnosti vstupů jsou ale zpracovávány stejně jako při vstupu/výstupu csv, kde data z databáze jsou prvně staženy do souborů csv a až poté zpracovány standardně.

## Příprava dat ve formátu CSV

Program Pregrouper na vstupu pracuje se 3 hlavními vstupními csv soubory pro doklady02, doklady06 a položky. Zde je nutné si uvědomit, že tyto soubory mohou dosahovat poměrně velkých velikostí, kde největším je vždy soubor s položkami. Z těchto velikostí vyplývají určité nutné kroky předpřípravy těchto dat, protože obvykle není možné je mít celé v paměti a něco v nich hledat.

### Seřazení vstupních souborů

Prvním krokem v procesu PgpMaster navíc oproti verzi v Talendu je seřazení všech vstupních dat. Implementace se nachází v třídě PgpOrderInputCsvFiles.

Doklady02 a doklady06 jsou všechny řádky řazeny podle abecedy, tedy primárně podle prvního sloupce ID_POJ (id pojištěnce) a položky jsou řazeny podle sloupce ID_DOKLADU. Důvod, proč zrovna podle těchto sloupců, bude zmíněn v pozdější části dokumentace.

Seřazení sloupců má různý typ řadícího algoritmu podle toho, zda se soubor vejde či nevejde celý do poskytnuté paměti běhového prostředí Java (JRE). Pokud se vejde, načte se celý do paměti a celý se v paměti i seřadí standardním způsobem.

Pokud se soubor celý do paměti nevejde, je nutné použít jiný řadící algoritmus. V aktuální verzi je použit external-merge-sort s komplexitou O(n*log n). Prozatím je řazení řízeno jedním vláknem a jeho rychlost zavisí zásadně na rychlosti disku, kvůli velkému množství I/O operací.

V rychlosti external-merge-sort funguje na jednoduchém principu, kde se vždy načte 1 část souboru, která se stále vejde do poskytnuté paměti JRE, ta se seřadí a uloží na disk jako samostatný soubor. Toto se dělá, dokud se nepřečte a nerozdělí původní soubor celý a následuje druhá fáze algoritmu.

V druhé fázi se z každé části přečte 1 řádek (pro omezení I/O operací se načte na pozadí naráz větší množství), tyto řádky se seřadí a první se zapíše do výsledného souboru. Ten řádek, který se zapsal, se nahradí dalším z příslušného souboru a proces se opakuje, dokud se všechny řádky ze všech rozdělených souborů nepřečtou. Po dokončení se smaže složka s pomocným soubory a ponechá se pouze výsledný seřazený.

### Příprava metadat pro random-access-file (RAF)

Po seřazení dat je potřeba připravit pomocná data pro soubor položek. Proces vyžaduje velké množství vyhledávání položek, které jsou navázané na konkrétní ID_DOKLADU, proto z hlediska výkonu není možné tento velký soubor pokaždé celý číst znovu, když je potřeba nalézt položky pro nějaký doklad.

Pro urychlení tohoto procesu vyhledávání je v programu využíváno RAF v Javě, která umožňuje skočit na konkrétní byte v souboru a začít číst od něj, bez nutnosti číst všechny byty před ním. Díky tomu se celý proces výrazně zrychlí. Aby toto bylo možné, je potřeba si připravit metadata o souboru položek, který bude obsahovat adresu příslušného byte, pokud je třeba najít nějaký konkrétní ID_DOKLADU.

V kódu jsou tato metadata nazývané „offsetShard“ a jsou ukládaná do stejnojmenné složky. Každý „offsetShard“ soubor obsahuje sadu metadat pro větší množství ID_DOKLADU a je pojmenovaný podle prvních 4 čísel ID_DOKLADU, pro které obsahuje data. Metadata v každém shardu jsou organizovaná do skupin 18 bytů, kde prvních 8 bytů reprezentuje hodnotu ID_DOKLADU, druhých 8 bytů reprezentuje index, kam RAF má skočit, aby přečetl toto ID_DOKLADU, a poslední 2 byty reprezentují počet řádků, které toto ID_DOKLADU obsahují. Toto uspořádání je možné jen díky předchozímu seřazení souboru položek podle ID_DOKLADU. Pak je snadné a relativně rychlé i ve velkých datech hledat potřebné doklady.

## Paralelní zpracování datových dávek

Na rozdíl od původních verzí v Talendu je Java verze Pregrouperu schopná zpracovávat data paralelně. To je díky předchozímu seřazení souborů doklady02 a doklady06 podle abecedy, resp. podle prvního sloupce ID_POJ. Toto seřazení umožňuje data rozdělit podle pojištěnců a každému vláknu přidělit datovou sadu obsahující vždy všechny doklady02, doklady06 a položky pro jednotlivé pojištěnce. Rychlost nahrávání těchto dat pro jednotlivá vlákna i přes všechny optimalizace prozatím zůstává hlavním omezením pro rychlost běhu programu.

## Proces agregace dat do hospitalizačních případů

Proces agregace dat probíhá ve velké míře v každém vlákně stejně jako v původním Talendu. V kódu jsou třídy pojmenované dle Talend dokumentace pro snadnou orientaci v jednotlivých krocích. Lze zde najít třídy s názvy např. PgpValidaceMaster, Pgp04bPolozkyCiselniky a Pgp05SestaveniVstupniVety. Některé kroky jsou ale upravené nebo přesunuté jinam v případě, že by již nedávaly smysl v nové struktuře programu. Jako příklad je možné uvést krok PgpDokladyValidaceK1, který přiřazuje chybové kódy podle chybějících/nevyhovujících vstupních dat. Zde některé chybové kódy popisují stav hodnot, které jsou ale vypočteny/dostupné až v posledním kroku Pgp05SestaveniVstupniVety. Proto jsou některé chybové kódy vyhodnoceny až v tomto kroku.

## Porovnání výstupu s Talend verzí

Po dokončení zpracování všech dokladů je možné zkontrolovat správnost výstupních hospitalizačních případů porovnáním s poskytnutým výstupem z Talendu. Zde je počítáno s tím, že se oba soubory vejdou do poskytnuté paměti JRE, dostupná paměť musí odpovídat velikosti vstupů.

Data z obou souborů jsou nejprve rozděleny podle ID_POJ a párovány podle sloupců *ID_POJ, ID_ZP, IDZZ, ODB_PRI, ODB_PRO, DATUM_PRI, DATUM_PRO.* Páry řádku jsou pak porovnávány podle hodnot v jejich sloupcích s několika výjimkami.

1. Sloupec ID_PRIPADU je ignorován protože v obou programech je tento sloupec používán jako počitadlo počtu zpracovaných případů. Jeho hodnota nijak nevychází ze vstupních dat a je tudíž závislá pouze na náhodném pořadí zpracování, hodnoty tedy budou vždy jiné.
2. Sloupec UPV v Talendu obsahuje prázdnou hodnotu a v Javě obsahuje hodnotu 0.
3. Sloupec VEKDEN v Talendu obsahuje hodnotu 365 a v Javě obsahuje hodnotu 0.

# Import projektu pre-grouper v nástroji NetBeans

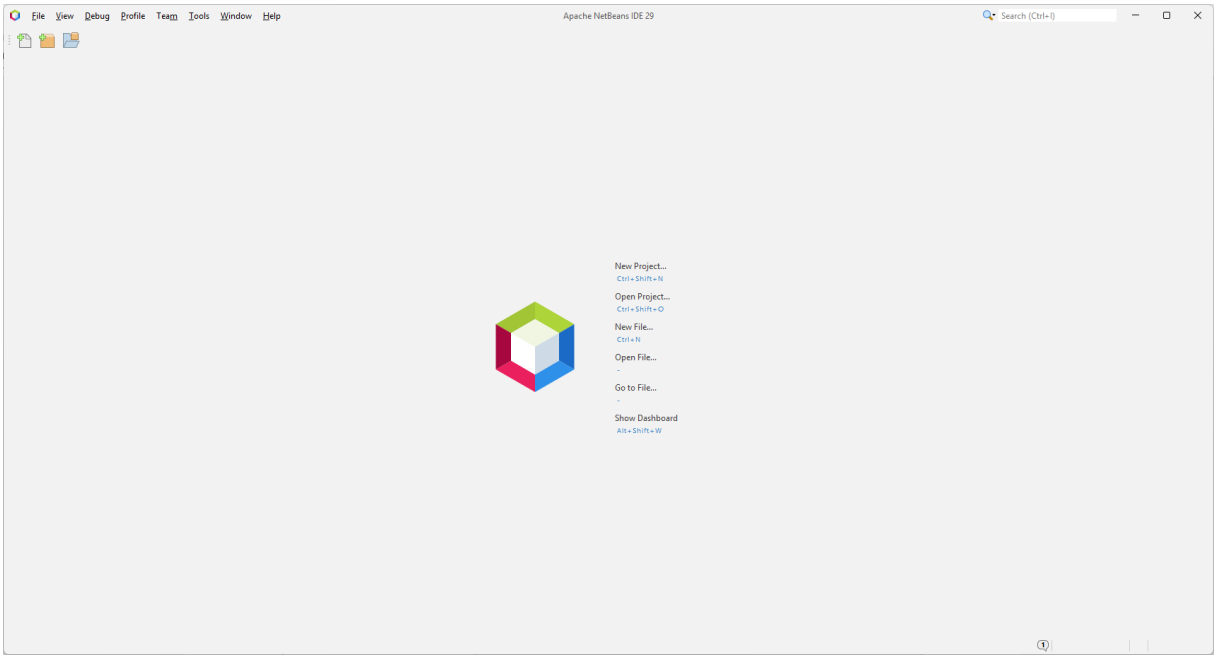
Součástí distribuce pre-grouperu jsou i kompletní zdrojové kódy tohoto nástroje. Kromě vlastní implementace metodiky sestavení hospitalizačního případu provádí obslužné činnosti nástroje – načtení konfigurace, vstupních dat i modelu, připojení do databáze, rozdělení vstupních dat do definovaných bloků, spuštění klasifikátoru, uložení výstupu apod. Projekt je možno načíst do vývojového prostředí, kompilovat, prohlížet nebo upravovat.

Před samotným nahráním souborů je potřeba program stáhnout a nainstalovat.

## Instalace nástroje NetBeans

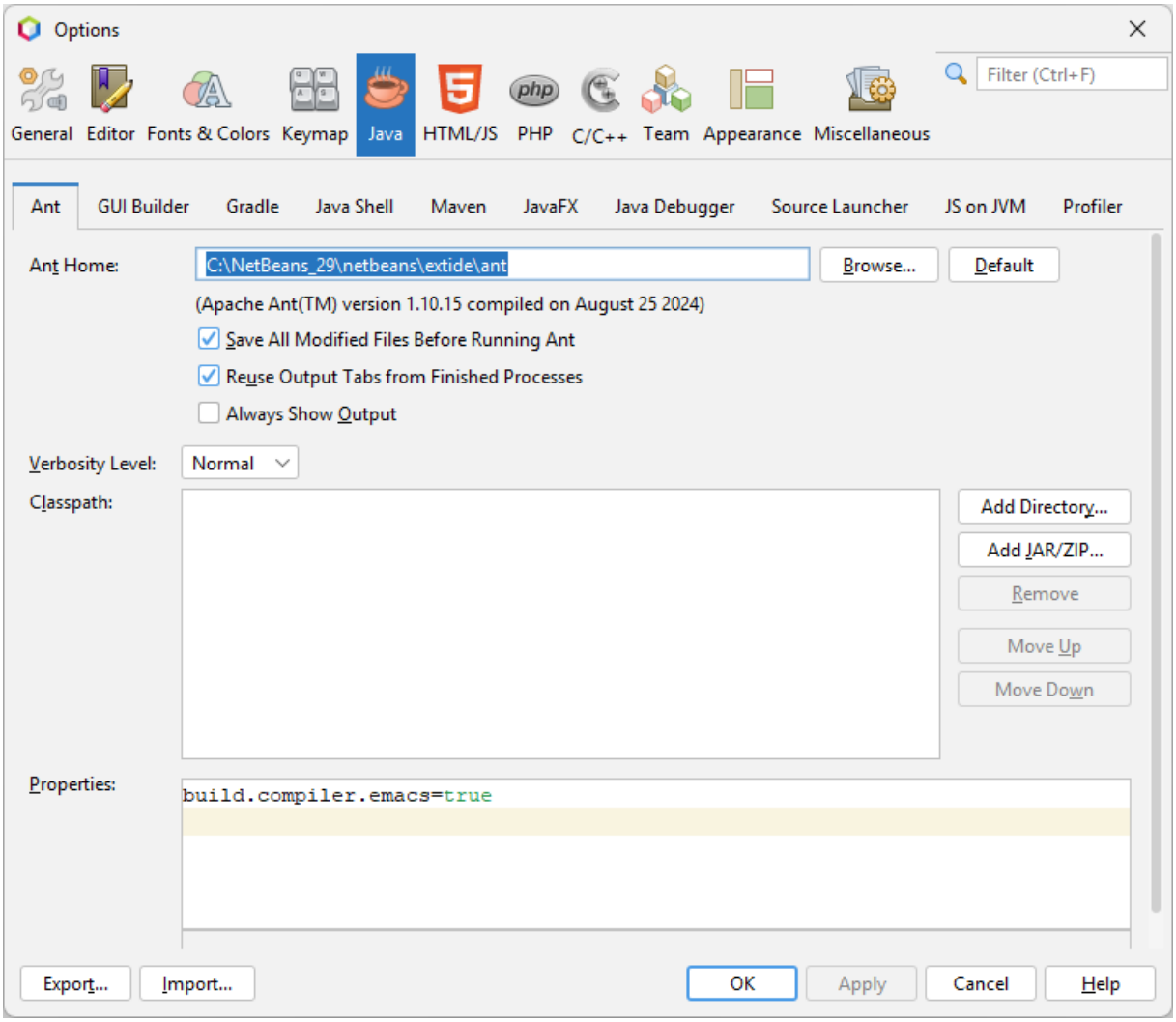
Vývojová platforma Apache NetBeans je dostupná ke stažení zdarma na adrese https://netbeans.apache.org/download/index.html. V době psaní aktualizace této dokumentace je k dispozici poslední verze 29 a verze starší, stačí stáhnout verzi Binary.

Po instalaci a spuštění platformy by se měla objevit úvodní obrazovka vývojového prostředí.




**Obrázek 1: Úvodní obrazovka Netbeans**

Potom je potřeba v nabídce Tools zvolit možnost Options a proklikat anebo případně upravit záložky týkající se SW Java.

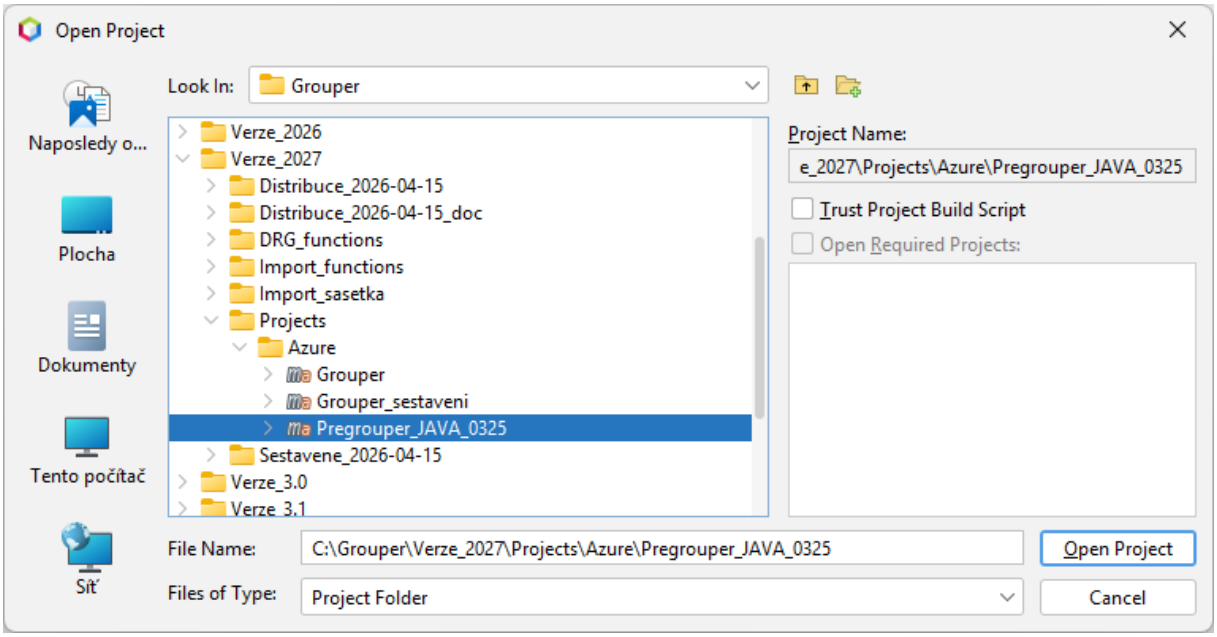



**Obrázek 2: Nastavení SW Java**

## Otevření projektu pro pre-grouper CZ-DRG

Dále je nutné stáhnout a rozbalit zdrojový kód nástroje Pregrouper do vybrané složky, např. *C:\Grouper\Projects\CZ-DRG-Grouper*.

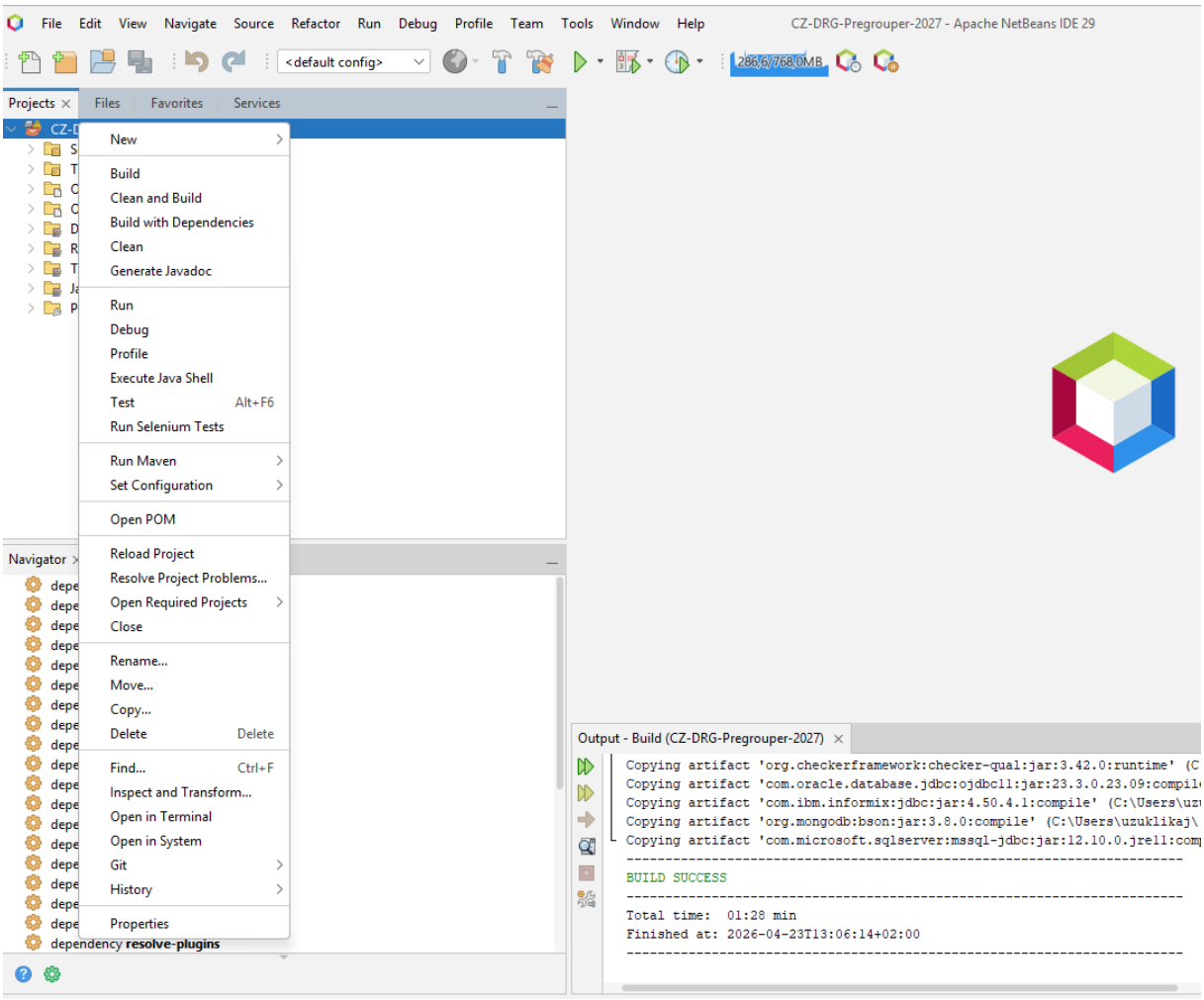
Pro otevření projektů lze použít nabídku uprostřed okna anebo tlačítko ***„Open Project...“*** , které se nachází v levém horním rohu nástroje (třetí zleva).




**Obrázek 3: Otevření projektu**

Ikona projektu signalizuje, že projekt používá pro sestavování nástroj Apache Maven, což znamená, že všechny knihovny nutné pro běh programu jsou stahovány automaticky při sestavení projektu, pokud nejsou nalezeny v lokálním úložišti Maven, a není nutné je distribuovat současně s projektem. Je však nutné mít internetové připojení pro úspěšné sestavení.

Pro spuštění procesu sestavení projektu je potřeba kliknout pravým tlačítkem na projekt v levé části NetBeans a následně na ***„Clean and Build“*** anebo využít tlačítko na liště nahoře (kladivo a štětec).




**Obrázek 4: Spuštění sestavení projektu**

Při úspěšném sestavení projektu by se měl objevit nápis „BUILD SUCCESS“ v konzoli NetBeans.

Pomocí průzkumníka souborů otevřete složku s projektem a podsložku **“target”**, kde se nachází sestavený projekt. Pro běh nástroje jsou potřebné pouze složky **“lib”, “settings”** a soubor .jar, které je možné zkopírovat a používat podle návodu v dokumentu INSTALAČNÍ PŘÍRUČKA A MANUÁL PRO UŽIVATELE NÁSTROJE CZ-DRG PRE-GROUPER.

## Popis automaticky generované dokumentace procesu pre-grouper

Všechny části procesu pre-grouperu jsou detailně dokumentovány automaticky generovanou dokumentací nástroje ~~Talend~~javadoc, která je k dispozici v podsložce ***přílohy*** v souboru ***pregrouper-2027-javadoc.zip***. ~~Dostupná je v samostatné složce ***talend_dokumentace***, která je součástí tohoto balíku technické dokumentace pre-grouperu. Je vytvořena ve formě HTML stránek, takže je čitelná v každém webovém prohlížeči a vedou do ní odkazy z tohoto dokumentu. Je členěna po jednotlivých stránkách pro každý jednotlivý proces (*job*), který je součástí pre-grouperu (viz následující části).~~

~~Hlavní částí každé stránky s popisem procesu je interaktivní náhledový obrázek procesu. Obrázek se nachází v horní části stránky v podkapitole ***Preview Picture.*** Kliknutím na jednotlivé komponenty procesu v obrázku se uživatel přenese do části dokumentace s popisem všech parametrů komponenty a jejich vyplněných hodnot. Typy parametrů a jejich počet je různý pro každou komponentu.~~

~~V následující části jsou popsány typy digramů a logika skládání komponent v rámci datových a procesních toků. V druhé části této kapitoly je uveden příklad čtení automatické dokumentace pro komponentu *tMap*.~~

### ~~Základní členění procesu~~

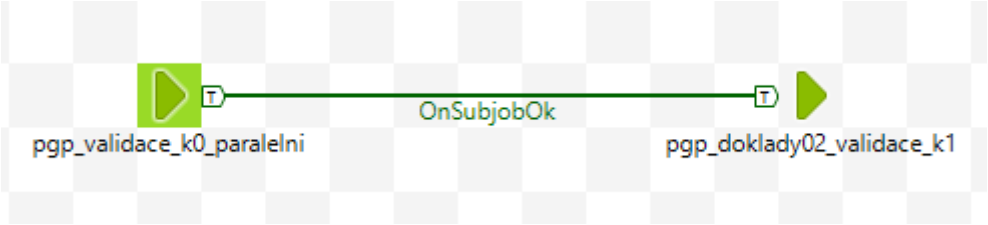
Celý proces pre-grouperu je z důvodu lepšího návrhu, dokumentace a použitelnosti strukturován do dílčích podprocesů, tzv. *„jobů“*, které jsou hierarchicky uspořádány od nadřazených celků k podřízeným jobům a od počáteční fáze procesu k jeho konci. Na pomyslném vrcholu této hierarchie je tzv. *„master job“* celého pre-grouperu, který spouští další části tohoto procesu (viz **PGP_SERVICE_MASTER**).

Každý *job* se skládá z jedné či více *komponent*, což jsou jednotlivé bloky procesu, které provádí nějakou jasně definovanou činnost, mají svoje vstupy a výstupy a mohou být parametrizovány. ~~Každé *komponentě* je přiřazena pojmenovaná grafická ikona v rámci diagramu, který popisuje daný *job*. Komponenty se dělí na datové a procesní a mohou se používat v rámci datových a procesních diagramů nebo i kombinovaně.~~

### ~~Procesní diagramy~~

~~*Procesní diagramy* popisují logiku a návaznost spouštění jednotlivých částí (komponent) jobu. Komponenty procesu jsou propojeny spojnicemi, které tuto logiku popisují. Typy spojnic jsou například *„po úspěšném ukončení předchozí akce“* (*OnSubjobOk*, *OnComponentOk*), *„po neúspěšném ukončení předchozí akce“* (*OnSubjobError*, *OnComponentError*) nebo podmínkové *if* spojnice pro definici vlastních podmínek. Komponenty pak spolu s jejich propojením tvoří tzv. *procesní diagramy* (*process-flow*). Zde je uveden příklad jednoduchého *procesního diagramu*, který znázorňuje spuštění dvou podprocesů, z nich druhý se spustí až po úspěšném ukončení prvního (viz **PGP_VALIDACE_MASTER**):~~

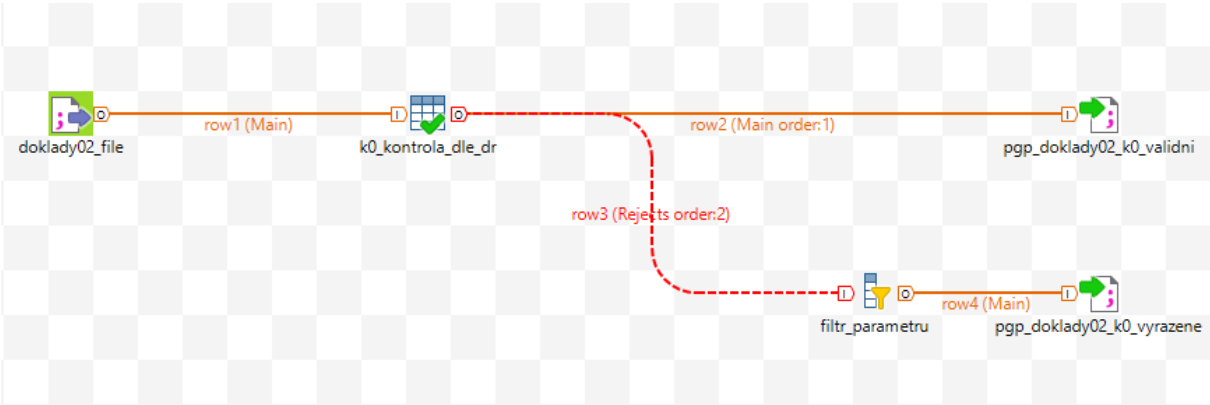
~~*Obrázek 1 Ukázka procesního diagramu*~~




## ~~Datové diagramy~~

~~Druhým typem jsou *datové diagramy*. Ty popisují průběh zpracování a jednotlivé transformace dat od vstupu až na výstup. *Komponenty* jsou propojeny spojnicemi, které značí přenos dat z jedné *komponenty* do druhé (*row1*, *row2*, *out1*, atd.). V závislosti na použití typu datové komponenty se můžou přesouvat všechny záznamy, záznamy splňující pouze zadanou podmínku, vybrané části záznamů apod. Dohromady celek navzájem spojených datových *komponent* tvoří *datový diagram* (*data flow*). Jednoduchý *datový diagram* je zobrazen na následujícím obrázku (viz také **PGP_DOKLADY02_VALIDACE_K0**):~~

~~*Obrázek 2 Ukázka datového diagramu*~~



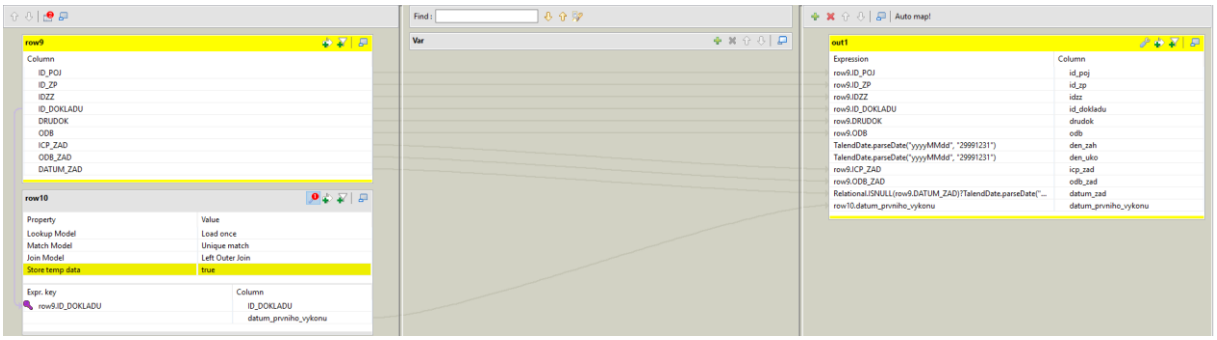

~~V tomto procesu jsou načítána data ze souboru ***doklady02_file*** a jsou přenášena větví row1 do komponenty ***k0_kontrola_dle_dr***, kde jsou kontrolována na datové typy, délky a vyplněnost. Na základě výsledku kontroly vycházejí data buď větví ***row2*** do souboru pgp_doklady02_k0_validní, nebo u nevalidních záznamů větví ***row3*** do komponenty ***filtr_parametru*** a následně jen vybrané parametry větví **row4** do souboru ***pgp_doklady02_k0_vyrazene***. Každá z komponent a jejich vazeb je detailně zdokumentována, viz následující část.~~

~~Oba typy diagramů (procesní a datový) s využitím procesních a datových komponent se v nástroji Talend mohou kombinovat, viz např. **PGP_DOKLADY06PRIPADY_EXTRAMURALNI**.~~

## ~~Ukázka popisu komponenty v generované dokumentaci~~

~~Nejkomplexnější komponentou nástroje datové integrace Talend v rámci použití i následné dokumentace je komponenta ***tMap*** . Ta umožnuje transformaci dat z jednoho či více vstupů do jednoho nebo více výstupů a jejich vzájemné *„mapování“* (přiřazení výstupních parametrů ke vstupním). Součástí dokumentace ***tMap*** komponenty je vždy náhledový obrázek, kde je toto *mapování* názorně zachyceno. Následující obrázek pochází z dokumentace procesu **PGP_02_DOKLADY**:~~

~~*Obrázek 3 Ukázka dokumentace komponenty tMap – mapování vstupních parametrů na výstupní*~~




~~Zde se spojují záznamy ze vstupů ***row9*** a ***row10*** na výstup, označený jako ***out1***.~~

~~V rámci parametrů této komponenty je zejména důležité sledovat *metadata* na výstupu. V části dokumentace ***Metadata Table Entries*** pro výstup z komponenty ***tMap*** (zde označený jako ***out1***) je ve sloupci ***Expression*** popsána logika transformace parametrů ze vstupu na výstup. Např. výstupní parametr ***id_poj*** je přiřazen přímo ze vstupního parametru ***row9.ID_POJ***, zatímco třeba výstupní hodnota parametru ***den_zah*** je vytvořena funkcí ***TalendDate.parseDate()***:~~

~~*Tabulka 1 Ukázka dokumentace komponenty tMap - Metadata Table Entries( out1 ):*~~

| ~~Name~~ | ~~Type~~ | ~~Expression~~ | ~~isNullable~~ |
|---|---|---|---|
| ~~id_poj~~ | ~~String~~ | ~~row9.ID_POJ~~ | ~~false~~ |
| ~~id_zp~~ | ~~Integer~~ | ~~row9.ID_ZP~~ | ~~true~~ |
| ~~idzz~~ | ~~int~~ | ~~row9.IDZZ~~ | ~~false~~ |
| ~~id_dokladu~~ | ~~String~~ | ~~row9.ID_DOKLADU~~ | ~~false~~ |
| ~~Drudok~~ | ~~String~~ | ~~row9.DRUDOK~~ | ~~false~~ |
| ~~Odb~~ | ~~String~~ | ~~row9.ODB~~ | ~~false~~ |
| ~~den_zah~~ | ~~java.util.Date~~ | ~~TalendDate.parseDate("yyyyMMdd", "29991231")~~ | ~~false~~ |
| ~~den_uko~~ | ~~java.util.Date~~ | ~~TalendDate.parseDate("yyyyMMdd", "29991231")~~ | ~~false~~ |
| ~~icp_zad~~ | ~~int~~ | ~~row9.ICP_ZAD~~ | ~~false~~ |
| ~~odb_zad~~ | ~~String~~ | ~~row9.ODB_ZAD~~ | ~~false~~ |
| ~~datum_zad~~ | ~~java.util.Date~~ | ~~Relational.ISNULL(row9.DATUM_ZAD) ? TalendDate.parseDate("yyyyMMdd", "29991231") : row9.DATUM_ZAD~~ | ~~false~~ |
| ~~datum_prvniho_vykonu~~ | ~~java.util.Date~~ | ~~row10.datum_prvniho_vykonu~~ | ~~false~~ |

~~Pokud na vstupu dochází k spojení datových toků (*join/lookup*), jsou informace o typu spojení uloženy v části dokumentace ***Mapper table Properties(<nazev_lookup_napojeni>)*** u připojovaného vstupu (datového toku), zde ***row10***. Například parametr *isInnerJoin* popisuje způsob napojení na řídící záznam („*left join*“, resp. „*inner join*“). Parametr *Matching mode* popisuje typ dané vazby (hodnota *UNIQUE MATCH* značí vazbu 1:1, hodnota *ALL_MATCHES* pak vazbu 1:N resp. M:N).~~

~~*Tabulka 2 Ukázka dokumentace komponenty tMap - Mapper table Properties( row10 ):*~~

| ~~Properties~~ | ~~Values~~ |
|---|---|
| ~~Name~~ | ~~row10~~ |
| ~~Matching mode~~ | ~~UNIQUE_MATCH~~ |
| ~~isMinimized~~ | ~~false~~ |
| ~~isReject~~ | ~~false~~ |
| ~~isRejectInnerJoin~~ | ~~false~~ |
| ~~isInnerJoin~~ | ~~false~~ |
| ~~expressionFilter~~ | ~~null~~ |

~~Pokud je v části ***Metadata table Properties(<nazev_vystupu>)*** nastaven parametr *isRejectInnerJoin* na hodnotu *true*, pak jsou na výstup předávány pouze ty záznamy, které se nepodařilo v rámci inner join napojit. Tento postup je využit např. v procesu **PGP_DOKLADY06PRIPADY_EXTRAMURALNI** pro vyloučení dokladů 06, které se mohou připojit jako intramurální (resp. nejasně přiřaditelná) péče k jinému případu.~~

~~Komponenta ***tMap*** má kromě vstupů a výstupů dále možnost definovat další pomocné proměnné. Definice těchto proměnných se vyhodnocují postupně tak, jak jsou proměnné v komponentě zapsané a jejich vypočítané hodnoty jsou v rámci data-flow přístupné dalšímu procházejícímu řádku na vstupu. Tato klíčová vlastnost je použita v procesu sestavení hospitalizačních případů z posloupnosti seřazených dokladů 02 (**PGP_01_HOSP_PRIPADY**) k uchování dopočítaných parametrů hospitalizačního případu, dokud nedojde k jeho uzavření (nalezení posledního dokladu 02 případu). V rámci dokumentace jsou přídavné proměnné popsané v části ***Metadata Table Entries(Var)***.~~

~~Klíčovým parametrem, který je dostupný v každé komponentě programu Talend je parametr ***Comment***. Tento parametr umožňuje vývojáři dokumentovat použití komponenty v rámci procesu za účelem lepšího pochopení implementované logiky.~~

~~Názvy jednotlivých parametrů komponent a jejich podrobný popis lze nalézt v online/offline dokumentaci nástroje Talend (viz **IMPORT PROJEKTU PRE-GROUPER V NÁSTROJI DATOVÉ INTEGRACE**).~~

~~Např. online dokumentace komponenty ***tFileInputDelimited*** pro načítání dat z textových (csv) souborů:~~

~~**HTTPS://HELP.TALEND.COM/READER/OG0CEVPPYA_LEK9VZKTKIG/319SSPWHIPR0VTZRAD~XOW**~~

~~Nebo dokumentace komponenty ***tFilterRow*** pro filtrování vstupního datového toku na základě zadané podmínky na hodnotu daného parametru nebo skupiny parametrů (řádková filtrace):~~

~~**HTTPS://HELP.TALEND.COM/READER/WFRS1ZABST3AZ8ZJBGOMSQ/BSANOKJHWNARRLIWLT6OJG**~~

## ~~Import projektu pre-grouper v nástroji datové integrace~~

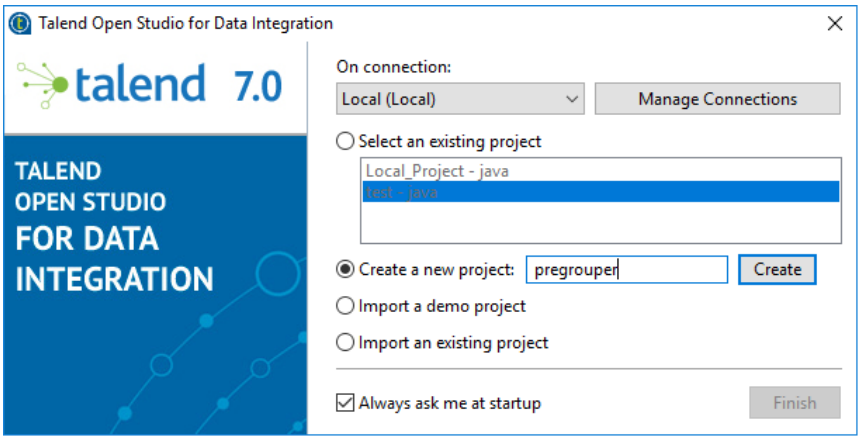
~~Před samotným nahráním souborů je potřeba program Talend Open Studio for Data Integration stáhnout a nainstalovat.~~

### ~~Instalace nástroje Talend~~

~~Nejnovější verzi programu lze stáhnout ze stránek společnosti Talend z následujícího odkazu (HTTPS://WWW.TALEND.COM/PRODUCTS/DATA-INTEGRATION/DATA-INTEGRATION-OPEN-STUDIO/). Program samotný se neinstaluje, stačí rozbalit stažený balík zip. V případě, že má uživatel administrátorská práva, je doporučeno program rozbalit do systémové složky (většinou C:\). V opačném případě je potřeba program rozbalit do své osobní složky v cestě C:\Users\<jmeno_uzivatele>\. V případě práce v prostředí Linux lze program rozbalit například v cestě /opt/.~~

~~Při spuštění programu je potřeba vytvořit a pojmenovat nový projekt:~~

~~*Obrázek 4 Vytvoření nového projektu v nástroji Talend*~~



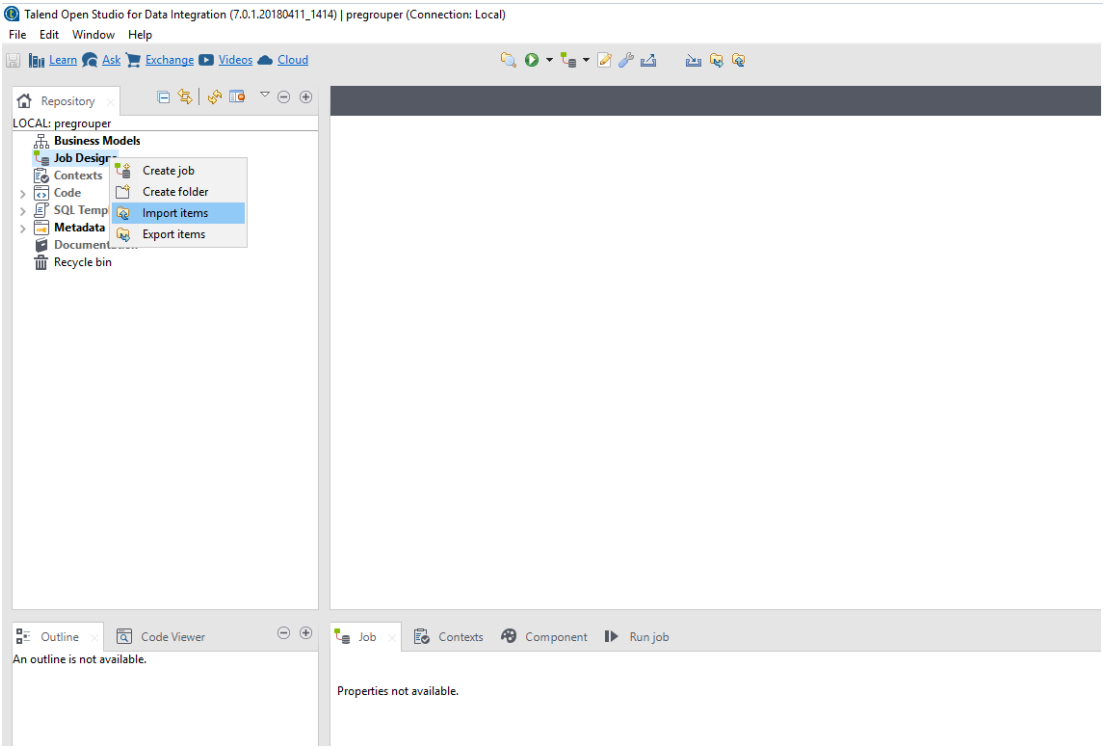

~~Po vytvoření projektu se při prvním spuštění zobrazí nabídka pro stažení a doinstalování potřebných knihoven třetích stran. Je potřeba odsouhlasit licenční podmínky (lze najednou pro balík všech knihoven) a knihovny doinstalovat.~~

~~V dalším kroku je třeba vytvořit nový projekt a nahrát do něj definici pre-grouperu z dodaného archivu.~~

~~Import projektu pre-grouperu~~

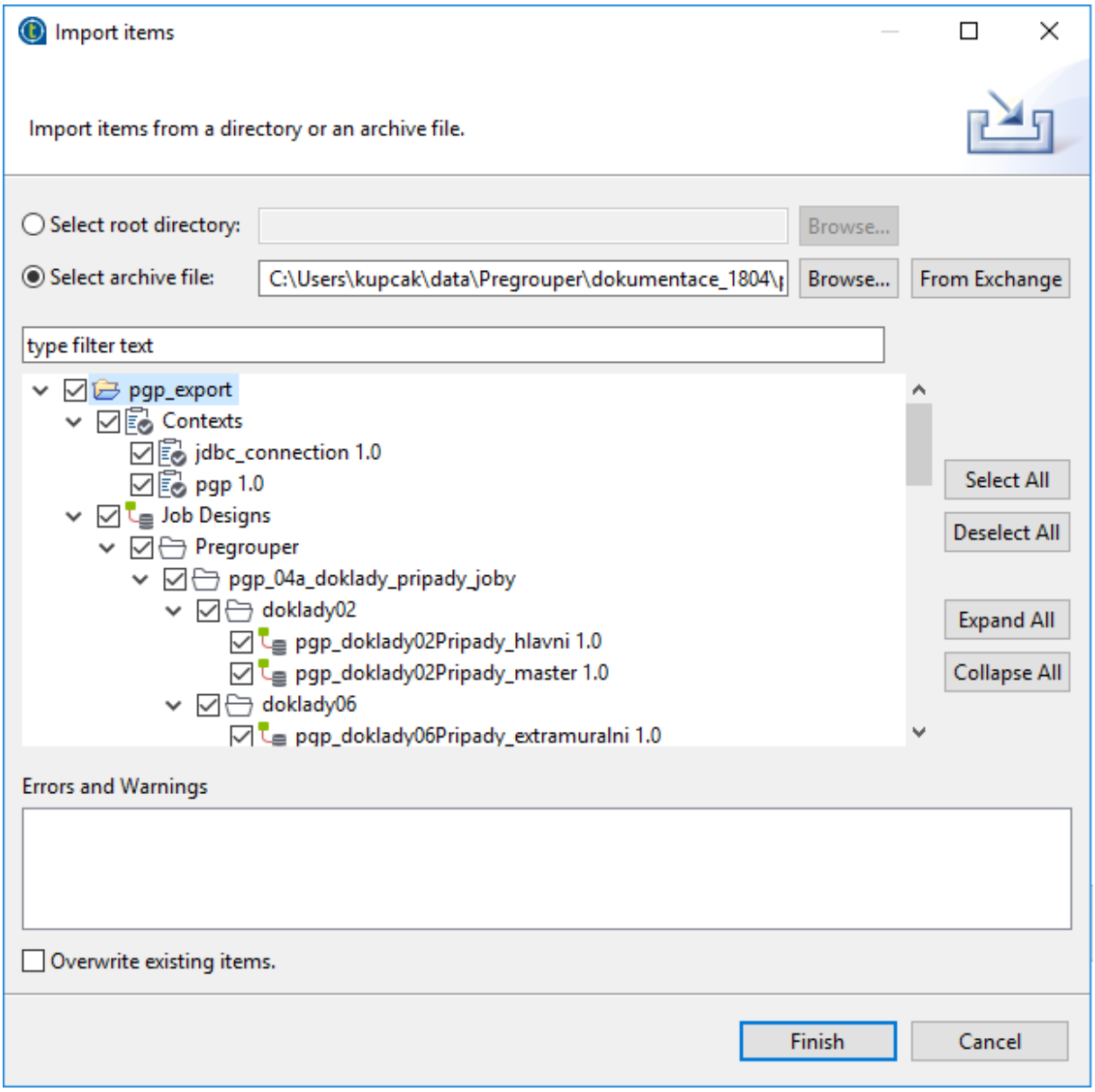
~~Nejprve stáhněte zveřejněný archiv 01__CZ-DRG_v7_r0_Pregrouper_Zdrojove_kody.zip a uložte jej na disk. Pro import definice pre-grouperu do programu Talend je potřeba klinout pravým tlačítkem myši na položku *Job Designs* v levém horním menu a z dostupných možností vybrat položku*Import items*.~~

~~*Obrázek 5 Import položek do nástroje Talend*~~



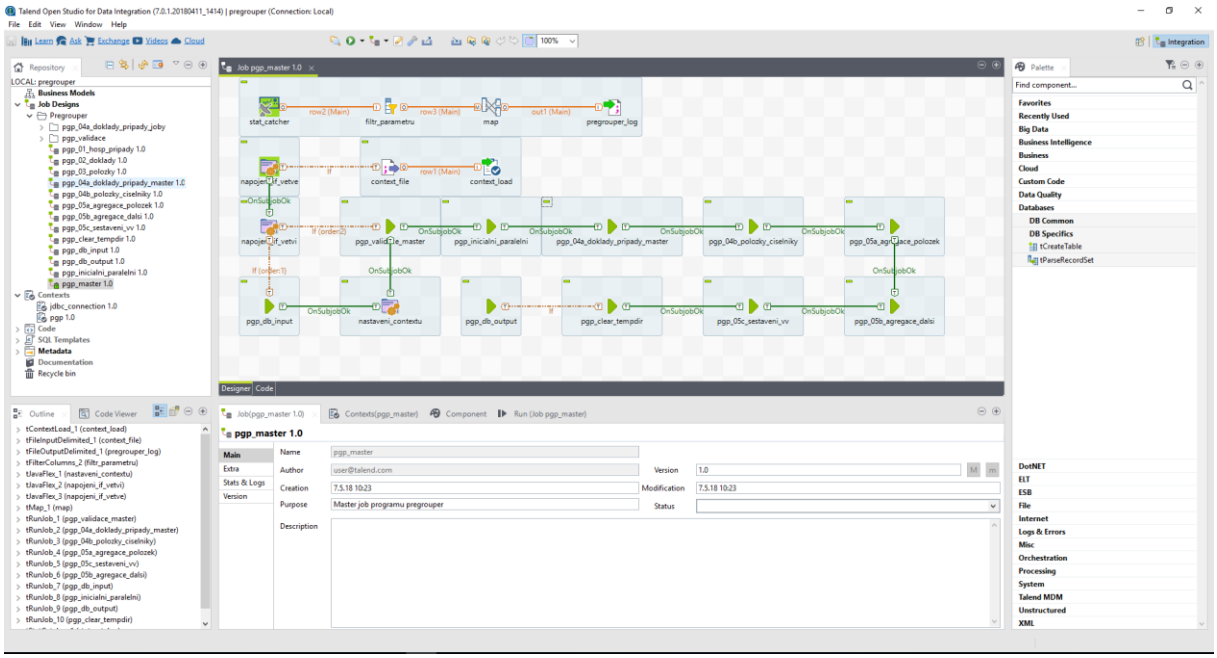
~~V následujícím okně vybrat možnost *Select archive file* pro import z komprimovaného souboru a vybrat soubor 01__CZ-DRG_v7_r0_Pregrouper_Zdrojove_kody.zip uložený na disku. Po načtení seznamu importovaných objektů zaškrtnout všechny položky a potvrdit tlačítkem *Finish*.~~

~~*Obrázek 6 Dokončení vytvoření projektu v nástroji Talend*~~



~~Po úspěšném načtení projektu se zobrazí pracovní plocha nástroje, ve které je možné s procesy dále pracovat, prohlížet si jednotlivé podprocesy a jejich komponenty atd.~~

~~*Obrázek 7 Pracovní plocha nástroje Talend*~~



## ~~Dokumentace nástroje Talend~~

~~K nástroji Talend Open Studio for Data Integration a možnostem použití jednotlivých komponent je k dispozici online a offline dokumentace.~~

~~Referenční manuál, uživatelská příručka a další dokumenty ve formátu PDF v komprimovaném balíku jsou k dispozici na následující adrese:~~

~~**HTTPS://WWW.TALEND.COM/PRODUCTS/DATA-INTEGRATION/DATA-INTEGRATION-MANUALS-RELEASE-NOTES/#PRODUCT_USER_MANUALS_ZIP**~~

~~Online dokumentace je k dispozici zde:~~

~~**HTTPS://HELP.TALEND.COM/**.~~

~~Z dostupných filtrů je potřeba vybrat produkt Talend Open Studio for Data Integration a příslušnou verzi produktu.~~

# Popis jednotlivých částí procesu pre-grouper

Následující kapitoly popisují jednotlivé části procesu pre-grouper (jedná se o základní popis funkčnosti odpovídající zpracování v Talendu).

## pgp_service_master

*Tabulka 3 Základní popis procesu pgp_service_master*

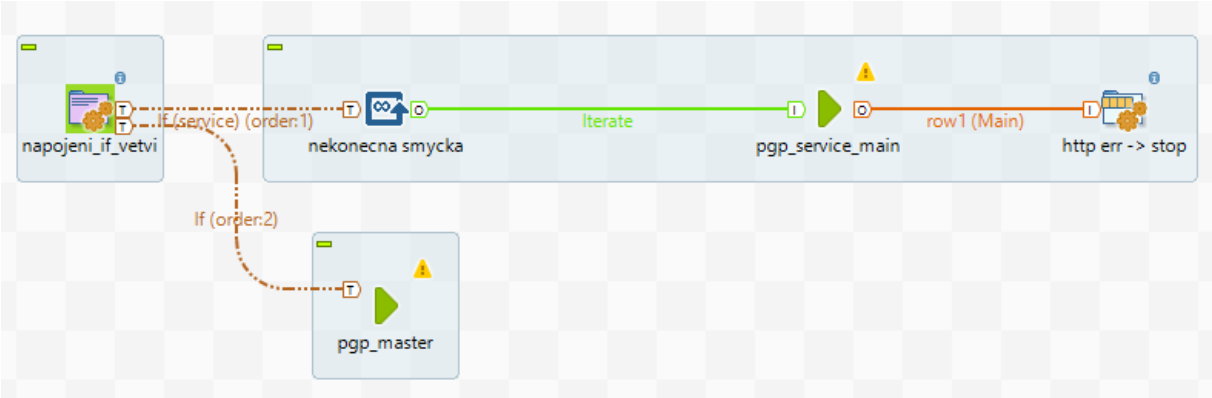
| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~**HTML – Talend dokumentace**~~ | PGP_SERVICE_MASTER | ~~talend_dokumentace\pgp_service_master\pgp_service_master_0.1.html~~ |
| **Nadřazený job** | - | master job celého pre-grouperu |
| **Předchozí job** | - | master job celého pre-grouperu |
| **Následující job** | PGP_MASTER | v případě jednorázového spuštění pregrouperu |
| **Následující job** | PGP_SERVICE_MAIN | v případě spuštění pregrouperu jako služby |
| **Vstupní data** | - | |
| **Výstupní data** | - | |

Nový master job (řídící proces) celého programu pre-grouper. Pomocí boolean kontextového parametru *service* zadaného při spuštění programu se definuje, jestli se pregrouper spouští jednorázově nebo jestli má běžet jako služba ~~(--context_param service=true)~~. Defaultní hodnota parametru *service* je false.

~~Do programu byla přidána nová kontextová skupina *pgp_service* s parametry:~~

- ~~*http_host*~~
- ~~*http_port*~~
- ~~*service*~~
- ~~*timer*~~

~~*Obrázek 8 Diagram procesu pgp_service_master*~~



## pgp_service_main

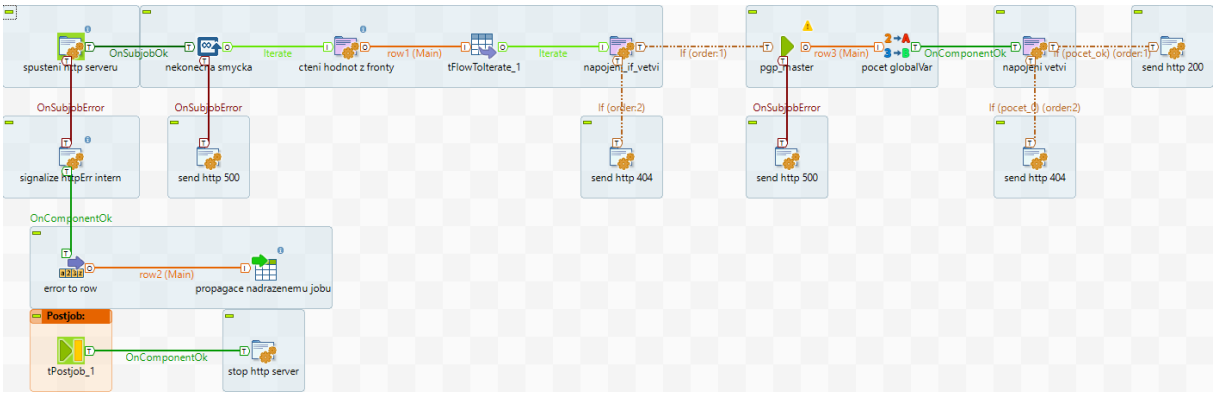
*Tabulka 4 Základní popis procesu pgp_service_main*

| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~HTML – Talend dokumentace~~ | PGP_SERVICE_MAIN | ~~talend_dokumentace\pgp_service_main\pgp_service_main_0.2.html~~ |
| Nadřazený job | PGP_SERVICE_MASTER | |
| Předchozí job | PGP_SERVICE_MASTER | |
| Následující job | PGP_MASTER | v případě spuštění pregrouperu jako služby |
| Vstupní data | - | |
| Výstupní data | - | |

Hlavní proces aplikace při spuštění jako služba. Spouští http server, na kterém poslouchá příchozí požadavky (zaslaný parametr runId). Ve smyčce iteračně spouští program s daným runId a v závislosti na dokončení vrací zaslanému požadavku koncový stav:

- 200: korektní zpracování a uložení dat do databáze
- 404: program nezpracoval žádná data (špatné runId, které se nevyskytuje v db)
- 500: interní chyba aplikace

~~*Obrázek 9 Diagram procesu pgp_service_main*~~



## pgp_master

*Tabulka 5 Základní popis procesu pgp_master*

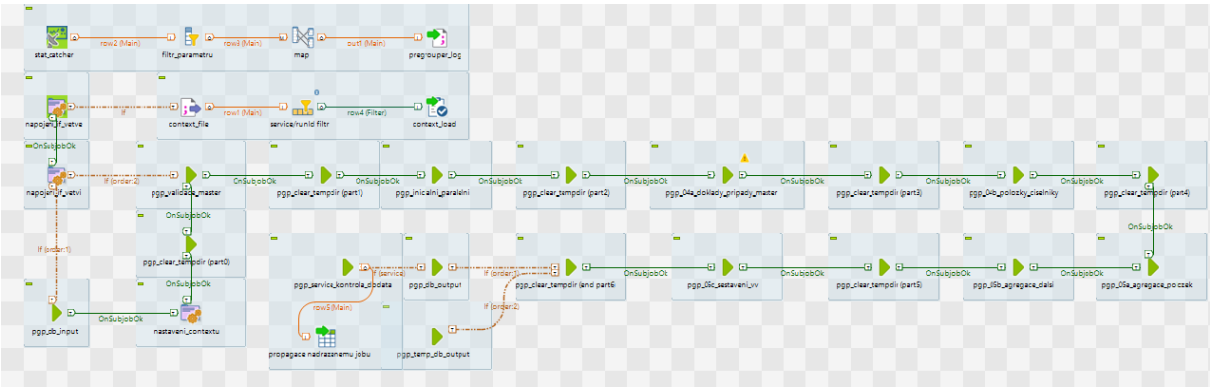
| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~**HTML – Talend dokumentace**~~ | PGP_MASTER | ~~talend_dokumentace\pgp_master\pgp_master_2.0.html~~ |
| **Nadřazený job** | PGP_SERVICE_MASTER | master job celého pre-grouperu |
| **Předchozí job** | PGP_SERVICE_MASTER | v případě jednorázového spuštění pregrouperu |
| **Předchozí job** | PGP_SERVICE_MAIN | v případě spuštění pregrouperu jako služby |
| **Následující job** | - | |
| **Vstupní data** | context_file | soubor s externí konfigurací |
| **Výstupní data** | pregrouper_log | logovací soubor pre-grouperu |

Původní master job (řídící proces) celého programu pre-grouper, hlavní proces zpracování v režimu jednorázového spuštění, který je využívaný i v režimu spuštění jako služba. Spouští jednotlivé podprocesy (subjoby) programu. Umožňuje načtení konfiguračního souboru zadaného při spuštění. V závislosti na nastavení konfiguračních parametrů se mohou spustit podprocesy načítání a ukládání z/do databáze nebo jen z/do souborů. Informace o čase spuštění a ukončení procesu, výsledném stavu a délce trvání se zapisují do logovacího souboru ~~v kořenové složce pre-grouperu~~.

~~Do kontextové skupiny *pgp* byla přidaná proměnná runId (ID spuštění), která je zpracovaná ve filtru u načítání kontextových proměnných z konfiguračního souboru. Proměnná runId případně definovaná v konfiguračním souboru se použije v případě jednorázového (dávkového) spuštění programu. Pokud běží program jako služba (parametr service je true), tak se přebírá hodnota parametru runId z nadřazeného procesu pgp_service_main.~~

~~Při spuštění v dávkovém režimu se nepovinná proměnná runId na vstupu z konfiguračního souboru uloží do výstupního souboru (vstupní datové věty programu Grouper) do parametru RUN_ID. Zpracují se pouze záznamy ze vstupních tabulek se shodným parametrem runId (pro zpracování všech záznamů se použije hodnota runId 0). V režimu služby se vkládá hodnota z nadřazeného procesu pgp_service_main.~~

~~*Obrázek 10 Diagram procesu pgp_master*~~



## pgp_db_input

*Tabulka 6 Základní popis procesu pgp_db_input*

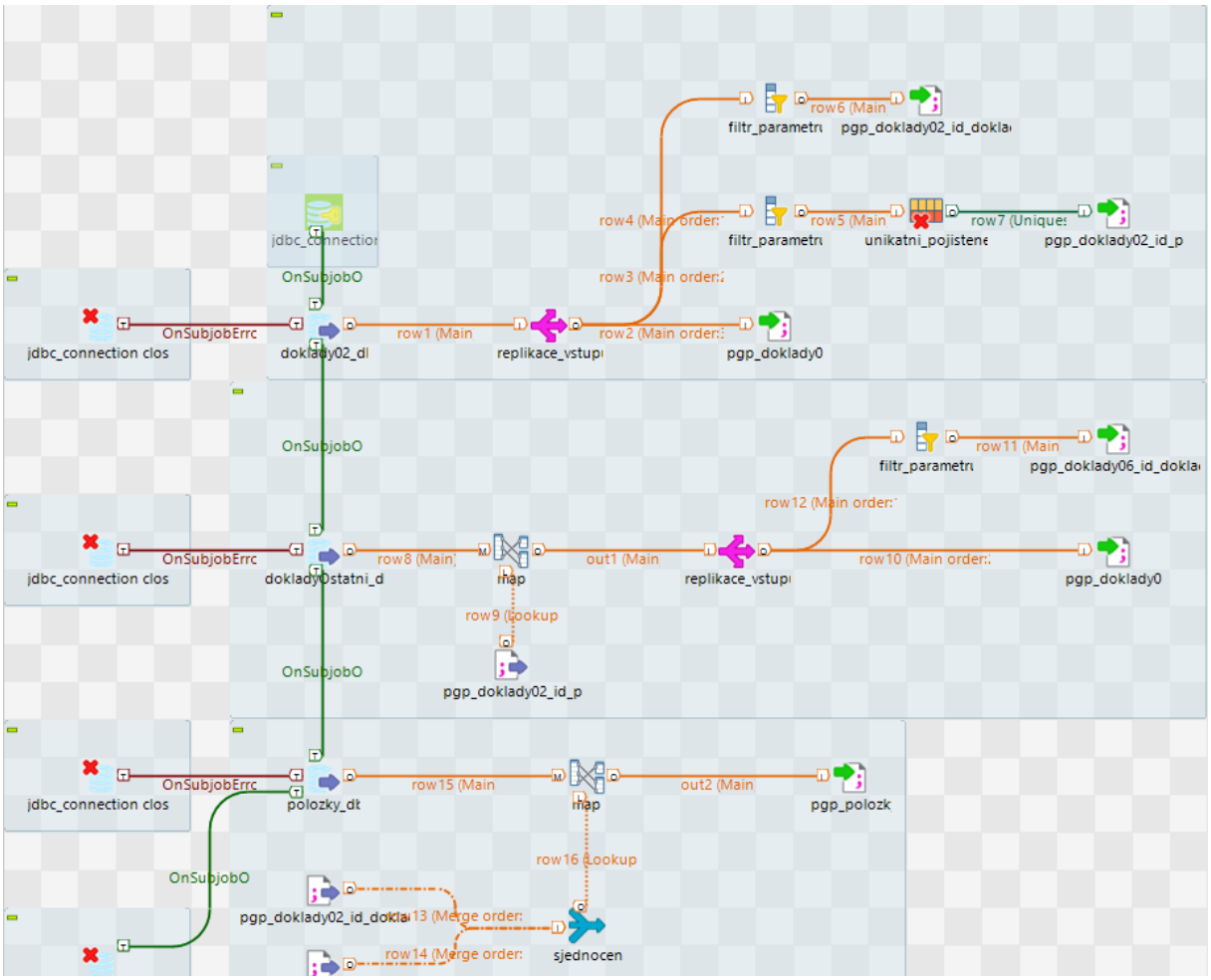
| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~**HTML – Talend dokumentace**~~ | PGP_DB_INPUT | ~~talend_dokumentace\pgp_db_input\pgp_db_input_1.3.html~~ |
| **Nadřazený job** | PGP_MASTER | |
| **Předchozí job** | - | |
| **Následující job** | PGP_VALIDACE_MASTER | |
| **Vstupní data** | doklady02_db | DB tabulka s hospitalizačními doklady |
| | dokladyOstatni_db | DB tabulka s poukazy na vyšetření |
| | polozky_db | DB tabulka s položkami dokladů |
| **Výstupní data** | pgp_doklady02 | načtené doklady 02 z databáze |
| | pgp_doklady02_id_dokladu | seznam id z načtených dokladů 02 (použije se jako vstup ve stejném procesu) |
| | pgp_doklady02_id_poj | seznam pojištěnců na dokladech 02 (použije se jako vstup ve stejném procesu) |
| | pgp_doklady06 | načtené doklady 06 z databáze |
| | pgp_doklady06_id_dokladu | seznam id z načtených dokladů 06 (použije se jako vstup ve stejném procesu) |
| | pgp_polozky | načtené položky dokladů z databáze |

Proces načtení vstupních dat z databáze. Po úspěšném připojení do databáze přes JDBC knihovnu je jako první zahájen proces načtení dokladů typu 02 z tabulky pgp_***doklady02_db***. Následuje načtení dokladů typu 06 z tabulky ***dokladyOstatni_db***, které jsou filtrovány pomocí unikátních identifikátorů pacientů z dokladů 02 (získaných v předchozím kroku – vybírají se pouze doklady 06 u pacientů, kteří mají alespoň jeden doklad 02). Posledním krokem je načtení položek z tabulky ***polozky_db***. Ty jsou filtrovány pomocí sjednoceného seznamu identifikátorů dokladů 02 a 06 (pouze položky z dokladů, vybraných v předchozích krocích). Data z načtených tabulek se ukládají do dočasných souborů ve složce <*tempdir*>.

Filtr vstupních dat byl rozšířený o využití parametru runId. Podmínka filtrace je upravena přes SQL funkci COALESCE (přebírá první hodnotu ze seznamu, která není NULL). Defaultní hodnota parametru runId pro zpracování všech dat z databáze je 0 (NULL > 0).

“WHERE COALESCE(RUN_ID, 0) = COALESCE(“+context.runId+”, 0)”

~~*Obrázek 11 Diagram procesu pgp_db_input*~~




Po úspěšném stažení všech tří souborů ~~si~~ proces ~~sám přenastaví hodnoty kontextových parametrů *indir*, *doklady02_file*, *dokladyOstatni_file* a *polozky_file* a~~ dále pracuje stejně jako se vstupem z textových souborů. Toto probíhá v nadřazeném procesu **PGP_MASTER**.

## pgp_validace_master

*Tabulka 7 Základní popis procesu pgp_validace_master*

| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~HTML – Talend dokumentace~~ | PGP_VALIDACE_MASTER | ~~talend_dokumentace\pgp_validace\pgp_validace_master\pgp_validace_master_1.1.html~~ |
| **Nadřazený job** | PGP_MASTER | |
| **Předchozí job** | PGP_DB_INPUT | pouze při čtení z DB |
| **Následující job** | PGP_VALIDACE_K0_PARALELNI | podřízený proces |
| **Vstupní data** | - | |
| **Výstupní data** | - | |

Master job validačních procesů vstupních datových souborů. V první fázi spouští paralelně validace typu K0 (validace v procesu PGP_VALIDACE_K0_PARALELNI vstupních parametrů podle datového rozhraní – datové typy, maximální délka, přesnost a vyplněnost) na vstupních souborech dokladů a položek. Následně spouští validace K1 (základní sémantické validace jednotlivých parametrů a jejich kombinací v procesu PGP_DOKLADY02_VALIDACE_K1) na souboru dokladů typu 02. Nakonec se spustí proces **PGP_VALIDACE_INFO**.

~~*Obrázek 12 Diagram procesu pgp_validace_master*~~

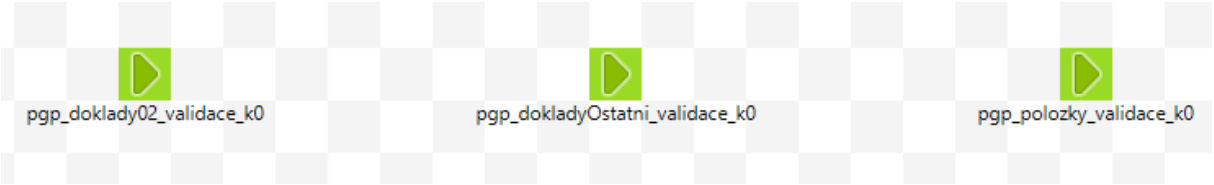
## pgp_validace_k0_paralelni

*Tabulka 8 Základní popis procesu pgp_validace_k0_paralelni*

| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~Dokumentace~~ | PGP_VALIDACE_K0_PARALELNI | ~~talend_dokumentace\pgp_validace\validace_k0\pgp_validace_k0_paralelni\pgp_validace_k0_paralelni_1.0.html~~ |
| **Nadřazený job** | PGP_VALIDACE_MASTER | |
| **Předchozí job** | PGP_VALIDACE_MASTER | |
| **Následující job** | PGP_DOKLADY02_VALIDACE_K0<br>PGP_DOKLADYOSTATNI_VALIDACE_K0<br>PGP_POLOZKY_VALIDACE_K0 | podřízené procesy spouštěné paralelně |
| **Vstupní data** | - | |
| **Výstupní data** | - | |

Řídící job procesu validace K0 vstupních souborů. Validaci souborů spouští paralelně v procesech PGP_DOKLADY02_VALIDACE_K0, PGP_DOKLADYOSTATNI_VALIDACE_K0 a PGP_POLOZKY_VALIDACE_K0.

~~Obrázek 13 Diagram procesu pgp_validace_k0_paralelni~~



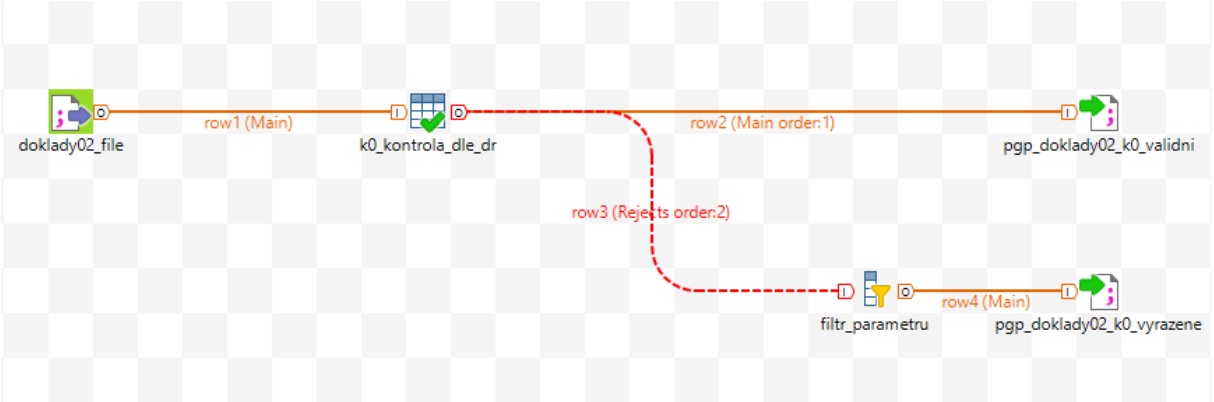

## *pgp_doklady02_validace_k0*

*Tabulka 9 Základní popis procesu pgp_doklady02_validace_k0*

| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~HTML – Talend dokumentace~~ | PGP_DOKLADY02_VALIDACE_K0 | ~~talend_dokumentace/pgp_validace/validace_k0/pgp_doklady02_validace_k0/pgp_doklady02_validace_k0_1.0.html~~ |
| **Nadřazený job** | PGP_VALIDACE_K0_PARALELNI | |
| **Předchozí job** | PGP_VALIDACE_K0_PARALELNI | nadřazený job |
| **Následující job** | PGP_DOKLADY02_VALIDACE_K1 | |
| **Vstupní data** | doklady02_file | soubor s hospitalizačními doklady, zadaný v rámci konfigurace nebo načtený z DB v procesu PGP_DB_INPUT |
| **Výstupní data** | pgp_doklady02_k0_validni | validní doklady 02, které prošly základní syntaktickou kontrolou K0 |
| | pgp_doklady02_k0_vyrazene | nevalidní doklady 02, které neprošly základní syntaktickou kontrolou K0 |

Validace typu K0 vstupního souboru dokladů typu 02. Parametry jsou načítány v textovém formátu s délkou 255 znaků a následně porovnávány oproti schématu parametrů nastavených dle datového rozhraní.

~~Obrázek 14 Diagram procesu pgp_doklady02_validace_k0~~




Záznamy s chybnými hodnotami jsou z procesu vyřazeny. Informace o identifikátoru dokladu a typu nalezené chyby (nalezených chyb) je uložena do dočasného souboru ***pgp_doklady02_k0_vyrazene***. Validní doklady jsou uloženy do souboru ***pgp_doklady02_k0_validni***.

*Tabulka 10 Validační soubory vytvořené v rámci procesu*

| Č. | Název | Cesta |
|---|---|---|

| | | |
|---|---|---|
| **1** | pgp_doklady02_k0_vyrazene | *<tempdir>*/validace_temp/k0/ vyrazene/pgp_doklady02_k0_vyrazene.csv |

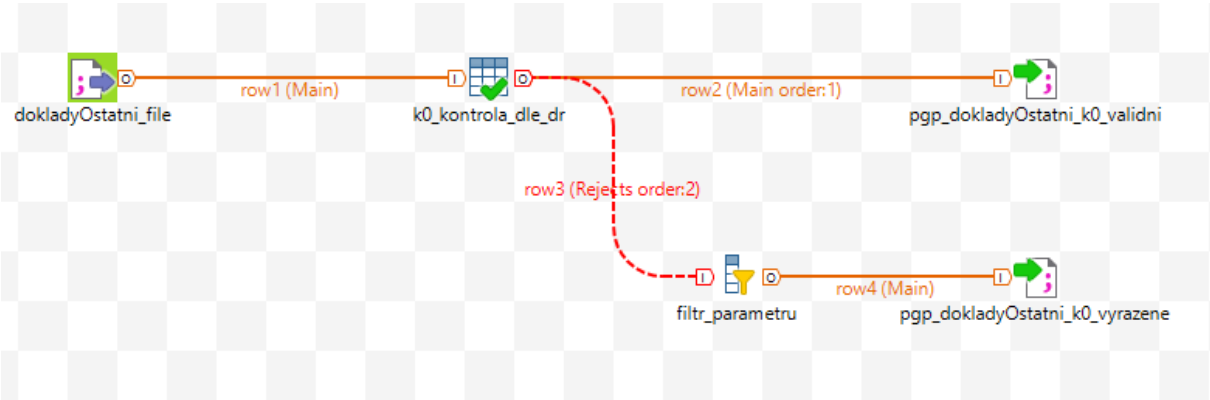
### *pgp_dokladyOstatni_validace_k0*

*Tabulka 11 Základní popis procesu pgp_dokladyOstatni_validace_k0*

| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~**HTML – Talend dokumentace**~~ | PGP_DOKLADYOSTATNI_VALIDACE_K0 | ~~talend_dokumentace\pgp_validace\validace_k0\pgp_dokladyOstatni_validace_k0\pgp_dokladyOstatni_validace_k0_1.0.html~~ |
| **Nadřazený job** | PGP_VALIDACE_K0_PARALELNI | |
| **Předchozí job** | PGP_VALIDACE_K0_PARALELNI | nadřazený job |
| **Následující job** | PGP_DOKLADY02_VALIDACE_K1 | |
| **Vstupní data** | dokladyOstatni_file | soubor s doklady poukazů, zadaný v rámci konfigurace, nebo načtený z DB v procesu PGP_DB_INPUT |
| **Výstupní data** | pgp_dokladyOstatni_k0_validni | validní doklady 06, které prošly základní syntaktickou kontrolou K0 |
| | pgp_dokladyOstatni_k0_vyrazene | nevalidní doklady 06, které neprošly základní syntaktickou kontrolou K0 |

Validace typu K0 vstupního souboru ostatních dokladů (tj. doklady 06 – poukazů na vyšetření/ošetření). Kontroly jsou analogií kontrol provedených v kroku PGP_DOKLADY02_VALIDACE_K0.

~~*Obrázek 15 Diagram procesu pgp_dokladyOstatni_validace_k0*~~




Záznamy s chybnými hodnotami jsou z procesu vyřazeny. Informace o identifikátoru dokladu a typu nalezené chyby (nalezených chyb) je uložena do dočasného souboru ***pgp_dokladyOstatni_k0_vyrazene***. Validní doklady jsou uloženy do souboru ***pgp_dokladyOstatni_k0_validni***.

*Tabulka 12 Validační soubory vytvořené v rámci procesu*

| Č. | Název | Cesta |
|---|---|---|
| **2** | pgp_dokladyOstatni_k0_vyrazene | *<tempdir>*/validace_temp/k0/ vyrazene/pgp_dokladyOstatni_k0_vyrazene.csv |

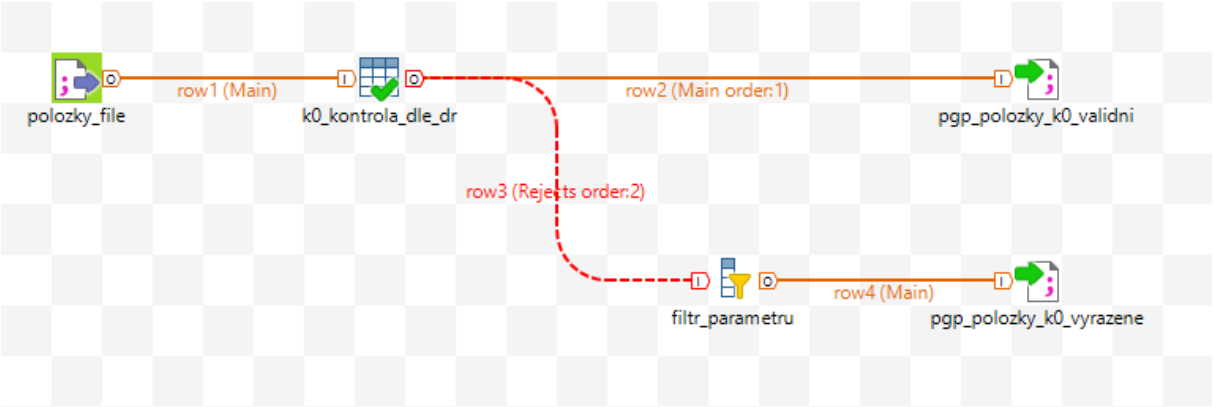
### *pgp_polozky_validace_k0*

*Tabulka 13 Základní popis procesu pgp_polozky_validace_k0*

| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~**HTML – Talend dokumentace**~~ | PGP_POLOZKY_VALIDACE_K0 | ~~talend_dokumentace\pgp_validace\validace_k0\pgp_polozky_validace_k0\pgp_polozky_validace_k0_1.0.html~~ |
| **Nadřazený job** | PGP_VALIDACE_K0_PARALELNI | |
| **Předchozí job** | PGP_VALIDACE_K0_PARALELNI | nadřazený job |
| **Následující job** | PGP_DOKLADY02_VALIDACE_K1 | |
| **Vstupní data** | polozky_file | soubor s položkami dokladů, zadaný v rámci konfigurace, nebo načtený z DB v procesu PGP_DB_INPUT |
| **Výstupní data** | pgp_polozky_k0_validni | validní položky dokladů, které prošly základní syntaktickou kontrolou K0 |
| | pgp_polozky_k0_vyrazene | nevalidní položky dokladů, které neprošly základní syntaktickou kontrolou K0 |

Validace typu K0 vstupního souboru položek dokladů 02 a 06. Kontroly jsou analogií kontrol provedených v kroku PGP_DOKLADY02_VALIDACE_K0.

~~*Obrázek 16 Diagram procesu pgp_polozky_validace_k0*~~




Záznamy s chybnými hodnotami jsou z procesu vyřazeny. Informace o identifikátoru dokladu a typu nalezené chyby (nalezených chyb) je uložena do dočasného souboru ***pgp_polozky_k0_vyrazene***. Validní doklady jsou uloženy do souboru ***pgp_polozky_k0_validni***.

*Tabulka 14 Dočasné soubory vytvořené v rámci procesu*

| Č. | Název | Cesta |
|---|---|---|
| **3** | pgp_polozky_k0_vyrazene | *<tempdir>*/validace_temp/k0/ vyrazene/pgp_polozky_k0_vyrazene.csv |

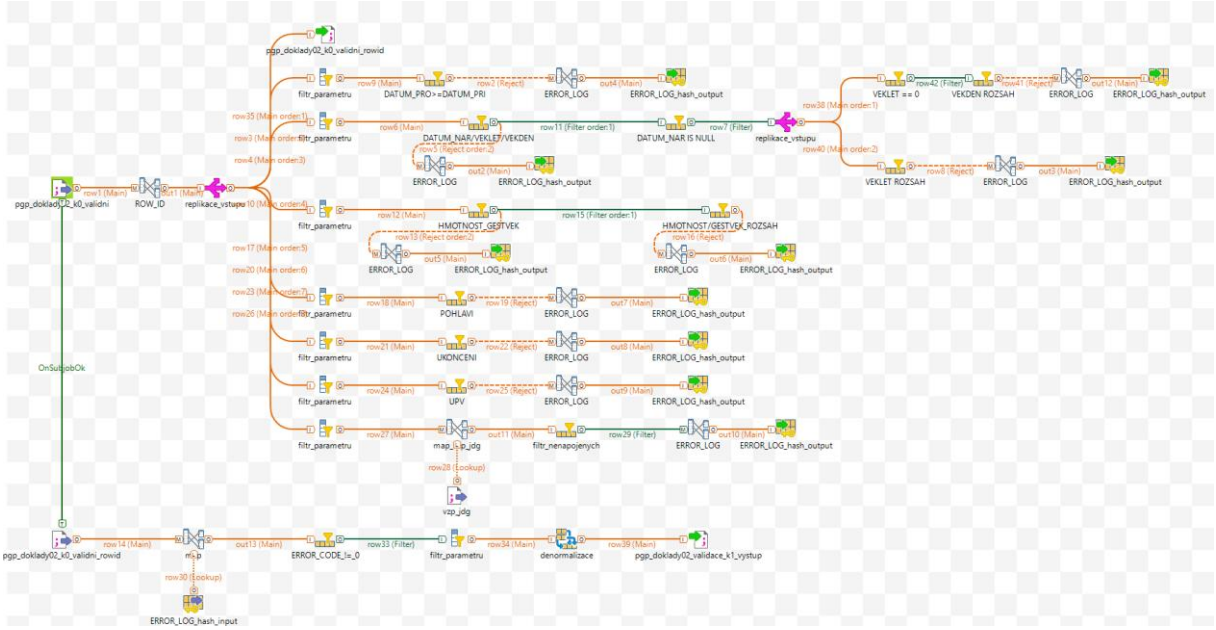
## pgp_doklady02_validace_k1

*Tabulka 15 Základní popis procesu pgp_doklady02_validace_k1*

| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~**HTML – Talend dokumentace**~~ | PGP_DOKLADY02_VALIDACE_K1 | ~~talend_dokumentace\pgp_validace\ pgp_doklady02_validace_k1\ pgp_doklady02_validace_k1_1.6.html~~ |
| **Nadřazený job** | PGP_VALIDACE_MASTER | |
| **Předchozí job** | PGP_DOKLADY02_VALIDACE_K0<br>PGP_DOKLADYOSTATNI_VALIDACE_K0<br>PGP_POLOZKY_VALIDACE_K0 | paralelně spouštěné validační procesy |
| **Následující job** | PGP_VALIDACE_INFO | |
| **Vstupní data** | pgp_doklady02_k0_validni | soubor validních dokladů 02 z procesu PGP_DOKLADY02_VALIDACE_K0 |
| | vzp_jdg | číselník diagnóz, které mohou být na dokladech 02 (načítá se ze složky *<ciselniky>*) |
| **Výstupní data** | pgp_doklady02_k0_validni_rowid | seznam ID řádků s doklady 02 ze vstupu; používá se v rámci stejného procesu pro vytvoření seznamu nalezených chyb |
| | pgp_doklady02_validace_k1_ vystup | seznam základních obsahových chyb K1, nalezený k jednotlivým řádkům dokladů 02 ze vstupu |

Proces provádí validace úrovně K1 na souboru dokladů typu 02, které úspěšně prošly validací K0 (***pgp_doklady02_k0_validni***). Vstupní záznamy prochází řadou kontrol s definovanými kódy chyb. Seznam kontrol odpovídá chybovým stavům, popsaným v datovém rozhraní pre-grouperu. Informace o nalezených chybách (kód chyby) na dokladech je společně se záznamy uložena do dočasného souboru ***pgp_doklady02_validace_k1_vystup***.

~~*Obrázek 17 Diagram procesu pgp_doklady02_validace_k1*~~




Žádný ze záznamů není na základě K1 kontrol vyřazen. Do dalšího zpracování postupují všechny záznamy dokladů 02, pouze jsou k výslednému HP přiřazeny chybové příznaky.

*Tabulka 16 Validační soubory vytvořené v rámci procesu*

| Č. | Název | Cesta |
|---|---|---|
| **4** | pgp_doklady02_validace_k1_vystup | *<tempdir>*/validace_temp/k1/ pgp_doklady02_validace_k1_vystup.csv |

## pgp_validace_info

*Tabulka 17 Základní popis procesu pgp_validace_info*

| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~**HTML – Talend dokumentace**~~ | PGP_VALIDACE_INFO | ~~talend_dokumentace\pgp_validace\ pgp_validace_info\ pgp_validace_info_0.1.html~~ |
| **Nadřazený job** | PGP_VALIDACE_MASTER | |
| **Předchozí job** | PGP_DOKLADY02_VALIDACE_K1 | |
| **Následující job** | PGP_INICIALNI_PARALELNI | |
| **Vstupní data** | pgp_polozky_k0_vyrazene | Soubor vyřazených položek dle validací K0 |
| | pgp_dokladyOstatni_k0_vyrazene | Soubor vyřazených dokladů ostatních typů dle validací K0 |
| | pgp_doklady02_k0_vyrazene | Soubor vyřazených dokladů typu 02 dle validací K0 |
| | pgp_doklady02_validace_k1_ vystup | Soubor nalezených chyb u dokladů 02 dle validačních kontrol K1 |

Proces kontroluje, zda existují výstupní soubory z kontrolních procesů K0 a K1. Informace o výskytu je vypsaná do příkazové řádky společně s cestou k danému souboru. Kontroly K1 na doklady typu 02 mají pouze informační charakter. Do procesu pokračují všechny doklady, které projdou kontrolou K0.

~~Obrázek 18 Diagram procesu pgp_validace_info~~

## pgp_inicialni_paralelni

*Tabulka 18 Základní popis procesu pgp_inicialni_paralelni*

| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~Dokumentace~~ | PGP_INICIALNI_PARALELNI | ~~talend_dokumentace\pgp_inicialni_paralelni\pgp_inicialni_paralelni_1.2.html~~ |
| Nadřazený job | PGP_MASTER | |
| Předchozí job | PGP_VALIDACE_INFO | |
| Následující job | PGP_DGVEDLEJSI_DEDUPLIKACE<br>PGP_01_HOSP_PRIPADY<br>PGP_02_DOKLADY<br>PGP_03_POLOZKY<br>PGP_VALIDACE_PRIPADU | podřízené procesy spouštěné paralelně; vyjímkou je proces pgp_dgvedlejsi_deduplikace spouštěný samostatně na začátku procesu a PGP_VALIDACE_PRIPADU, který se spouští samostatně na konci procesu |
| Vstupní data | - | |
| Výstupní data | - | |

Proces iniciálního zpracování vstupních validovaných souborů. Na začátku je spuštěn proces PGP_DGVEDLEJSI_DEDUPLIKACE, který odstraní případně existující duplicity ve vedlejších diagnózách.

Následně dochází k sestavení hospitalizačních případů v procesu PGP_VDG_ZAVAZNOSTI

TABULKA 20 Základní popis procesu pgp_vdg_zavaznosti

| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~HTML – Talend dokumentace~~ | PGP_VDG_ZAVAZNOSTI | ~~talend_dokumentace\pgp_vdg_zavaznosti\pgp_vdg_zavaznosti_0.1.html~~ |
| Nadřazený job | PGP_DGVEDLEJSI_DEDUPLIKACE | |
| Předchozí job | PGP_DGVEDLEJSI_DEDUPLIKACE | nadřazený job |
| Následující job | PGP_01_HOSP_PRIPADY<br>PGP_02_DOKLADY<br>PGP_03_POLOZKY | procesy spouštěné paralelně |

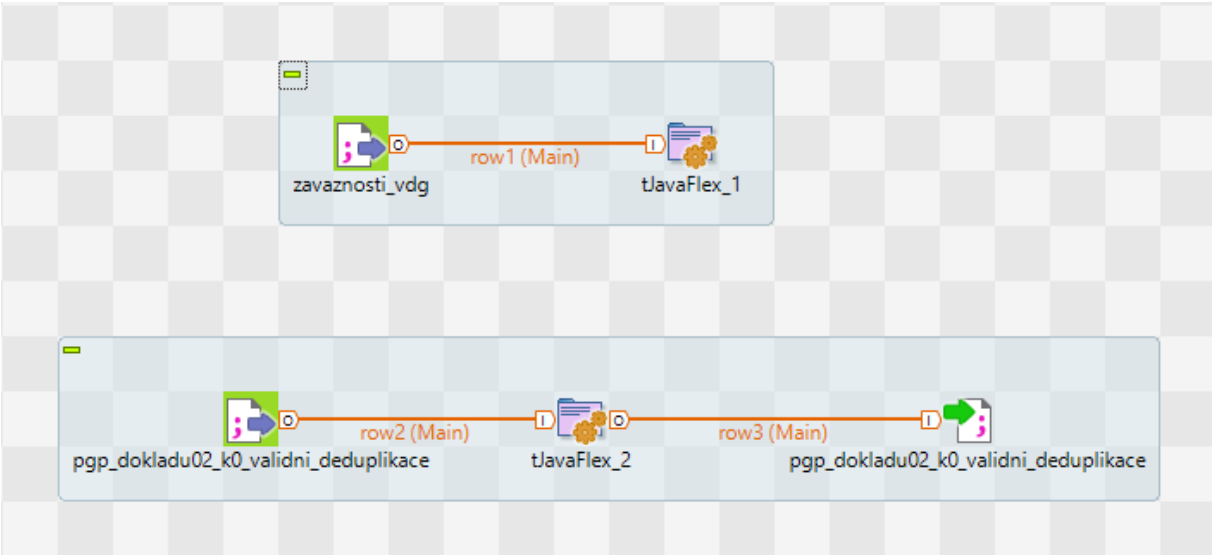
| | | |
|---|---|---|
| **Vstupní data** | zavaznosti_vdg | nový číselník od verze 7.0, obsahuje závažnosti skupin vedlejších diagnóz |
| | pgp_doklady02_k0_validni_deduplikace | soubor validních dokladů 02 s odstraněním případně existujících duplicit vedlejších diagnóz |
| **Výstupní data** | pgp_doklady02_k0_validni_deduplikace_vdg | soubor validních dokladů 02 s případným nastavením typu vedlejších diagnóz na hodnotu 9 |

Toto je nový funkční modul od verze 7.0. Prvním krokem je načtení vstupního souboru ***zavaznosti_vdg.csv*** (nový číselník obsahující 3 sloupce – skupinu vedlejších diagnóz, kód a závažnost). Dalším krokem je načtení souboru ***pgp_doklady02_k0_validni_deduplikace***. Potom se porovná všech ~~1430~~ sloupců vedlejších diagnóz DG_VEDLEJSIx každého záznamu s obsahem číselníku ***zavaznosti_vdg.csv*** a rozhodne se, zda se nastaví hodnota 9 do příslušného sloupce DG_VEDLEJSI_TYPx (účelem tohoto nastavení je, že při následné klasifikaci každé vstupní věty v Grouperu se bude považovat závažnost vedlejších diagnóz s typem 9 za nulovou).

Vyhodnocení se provádí tak, že se postupně pro každou skupinu vedlejších diagnóz z číselníku ***zavaznosti_vdg.csv*** projdou všechny kombinace vedlejších diagnóz, které jsou v daném záznamu a zároveň jsou ve zpracovávané skupině číselníku. Nejdříve se vybere ta vedlejší diagnóza, která má ve zpracovávané skupině nejvyšší závažnost. Jestliže jsou v daném záznamu ještě jiné vedlejší diagnózy ze zpracovávané skupiny, tak se nastaví všem vedlejším diagnózám daného záznamu do sloupce DG_VEDLEJSI_TYPx hodnota 9, pokud je jejich závažnost menší než nalezená maximální závažnost.

Po provedení všech změn na celých datech se výsledek zapíše do výstupního souboru ***pgp_doklady02_k0_validni_deduplikace_vdg***.

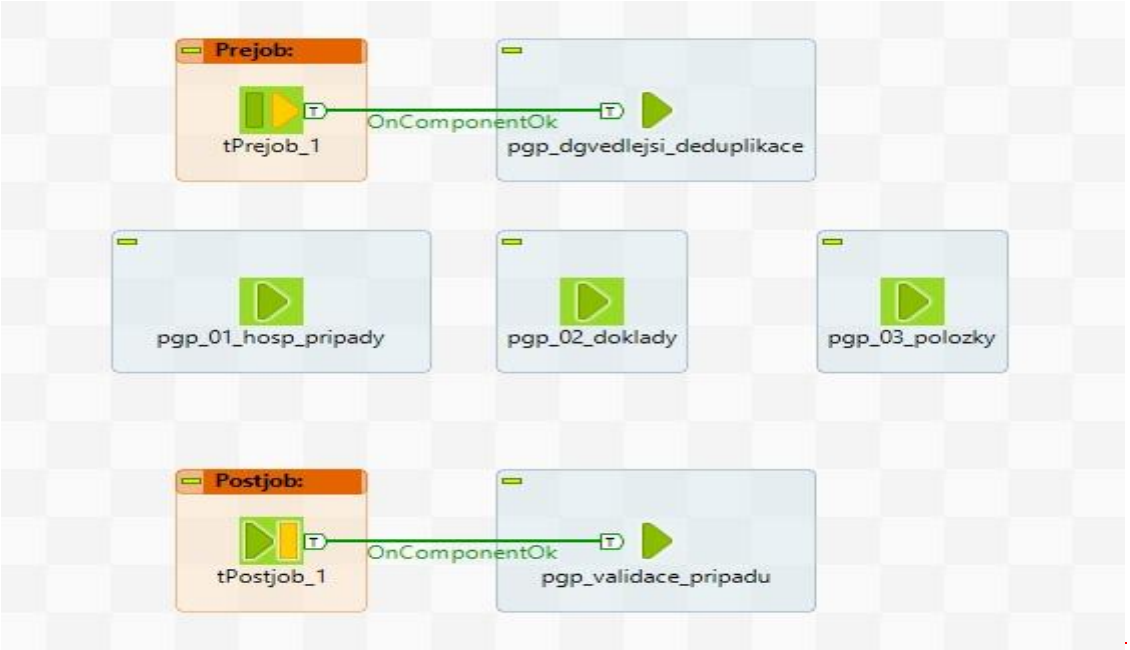
~~*Obrázek 21 Diagram procesu pgp_vdg_zavaznosti*~~




pgp_01_hosp_pripady, spojení dokladů obou typů v procesu PGP_02_DOKLADY a sjednocení sledovaných výkonů a DRG markerů a zvlášť účtovaných položek v procesu PGP_03_POLOZKY. Jednotlivé iniciální procesy probíhají paralelně.

Po dokončení paralelního běhu podprocesů se na závěr spouští proces PGP_VALIDACE_PRIPADU.

~~*Obrázek 19 Diagram procesu pgp_inicialni_paralelni*~~



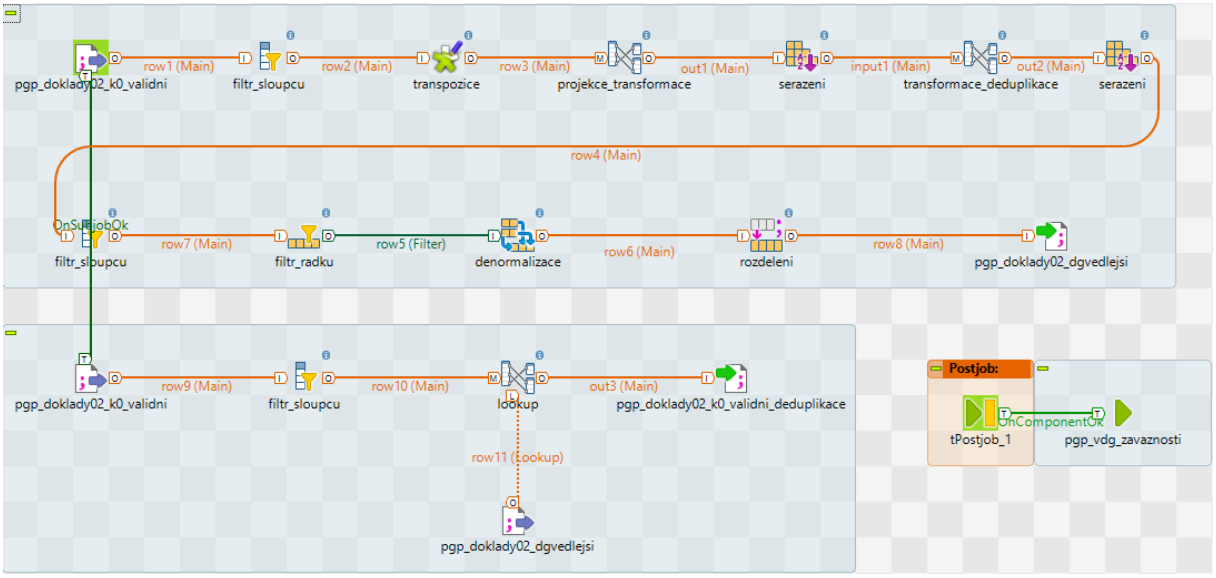

## pgp_dgvedlejsi_deduplikace

*Tabulka 19 Základní popis procesu pgp_dgvedlejsi_deduplikace*

| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~**HTML – Talend dokumentace**~~ | PGP_DGVEDLEJSI_DEDUPLIKACE | ~~talend_dokumentace\pgp_dgvedlejsi_deduplikace\pgp_dgvedlejsi_deduplikace_0.2.html~~ |
| **Nadřazený job** | PGP_INICIALNI_PARALELNI | |
| **Předchozí job** | PGP_INICIALNI_PARALELNI | nadřazený job |
| **Následující job** | PGP_VDG_ZAVAZNOSTI | |
| **Vstupní data** | pgp_doklady02_k0_validni | soubor validních dokladů 02 z procesu PGP_DOKLADY02_VALIDACE_K0 |
| **Výstupní data** | pgp_doklady02_k0_validni_deduplikace | soubor validních dokladů 02 s odstraněním případně existujících duplicit vedlejších diagnóz |

Proces odstraní případně existující duplicity ve vedlejších diagnózách dle upravené metodiky sestavení hospitalizačního případu. Vedlejší diagnózy rozdělí do řádků a setřídí, iteračně projde a vyřadí duplicity a opět je sestaví do jednoho řádku ke každému dokladu. Výstup ukládá do souboru *pgp_doklady02_k0_validni_deduplikace*.

~~Obrázek 20 Diagram procesu pgp_dgvedlejsi_deduplikace~~




## pgp_vdg_zavaznosti

*Tabulka 20 Základní popis procesu pgp_vdg_zavaznosti*

| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~HTML – Talend dokumentace~~ | PGP_VDG_ZAVAZNOSTI | ~~talend_dokumentace\pgp_vdg_zavaznosti\pgp_vdg_zavaznosti_0.1.html~~ |
| **Nadřazený job** | PGP_DGVEDLEJSI_DEDUPLIKACE | |
| **Předchozí job** | PGP_DGVEDLEJSI_DEDUPLIKACE | nadřazený job |
| **Následující job** | PGP_01_HOSP_PRIPADY<br>PGP_02_DOKLADY<br>PGP_03_POLOZKY | procesy spouštěné paralelně |
| **Vstupní data** | zavaznosti_vdg | nový číselník od verze 7.0, obsahuje závažnosti skupin vedlejších diagnóz |
| | pgp_doklady02_k0_validni_deduplikace | soubor validních dokladů 02 s odstraněním případně existujících duplicit vedlejších diagnóz |
| **Výstupní data** | pgp_doklady02_k0_validni_deduplikace_vdg | soubor validních dokladů 02 s případným nastavením typu vedlejších diagnóz na hodnotu 9 |

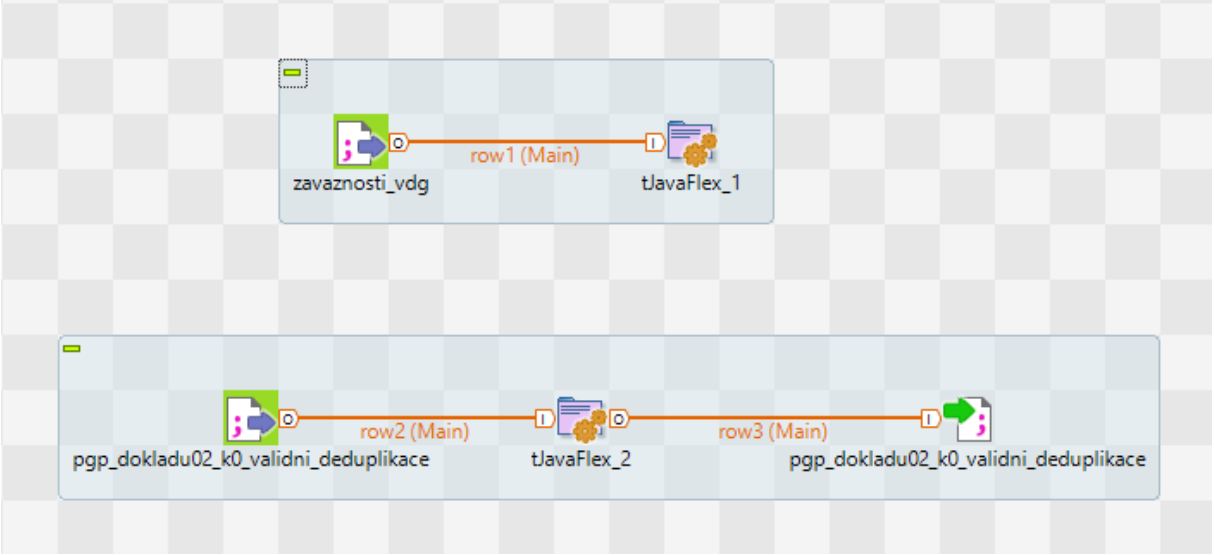
Toto je nový funkční modul od verze 7.0. Prvním krokem je načtení vstupního souboru ***zavaznosti_vdg.csv*** (nový číselník obsahující 3 sloupce – skupinu vedlejších diagnóz, kód a závažnost). Dalším krokem je načtení souboru ***pgp_doklady02_k0_validni_deduplikace***. Potom se porovná všech ~~1430~~ sloupců vedlejších diagnóz DG_VEDLEJSIx každého záznamu s obsahem číselníku ***zavaznosti_vdg.csv*** a rozhodne se, zda se nastaví hodnota 9 do příslušného sloupce DG_VEDLEJSI_TYPx (účelem tohoto nastavení je, že při následné klasifikaci každé vstupní věty v Grouperu se bude považovat závažnost vedlejších diagnóz s typem 9 za nulovou).

Vyhodnocení se provádí tak, že se postupně pro každou skupinu vedlejších diagnóz z číselníku ***zavaznosti_vdg.csv*** projdou všechny kombinace vedlejších diagnóz, které jsou v daném záznamu a zároveň jsou ve zpracovávané skupině číselníku. Nejdříve se vybere ta vedlejší diagnóza, která má

ve zpracovávané skupině nejvyšší závažnost. Jestliže jsou v daném záznamu ještě jiné vedlejší diagnózy ze zpracovávané skupiny, tak se nastaví všem vedlejším diagnózám daného záznamu do sloupce DG_VEDLEJSI_TYPx hodnota 9, pokud je jejich závažnost menší než nalezená maximální závažnost.

Po provedení všech změn na celých datech se výsledek zapíše do výstupního souboru ***pgp_doklady02_k0_validni_deduplikace_vdg***.

~~*Obrázek 21 Diagram procesu pgp_vdg_zavaznosti*~~




## pgp_01_hosp_pripady

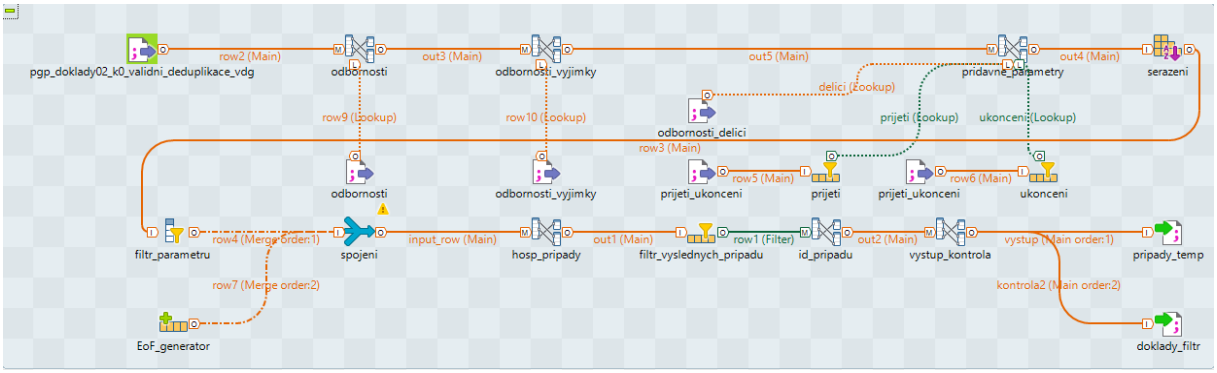
*Tabulka 21 Základní popis procesu pgp_01_hosp_pripady*

| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~**HTML – Talend dokumentace**~~ | PGP_01_HOSP_PRIPADY | ~~talend_dokumentace\pgp_01_hosp_pripady\pgp_01_hosp_pripady_2.0.html~~ |
| **Nadřazený job** | PGP_INICIALNI_PARALELNI | |
| **Předchozí job** | PGP_INICIALNI_PARALELNI | nadřazený job |
| **Následující job** | PGP_VALIDACE_PRIPADU | |
| **Vstupní data** | pgp_doklady02_k0_validni_deduplikace_vdg | soubor validních dokladů 02 z procesu PGP_VDG_ZAVAZNOSTI |
| | prijeti_ukonceni | číselník kódů přijetí a ukončení hospitalizace, seřazených dle pravidel *metodiky* (načítá se ze složky *<ciselniky>*) |
| | odbornosti | číselník prostředních hodnot zpracovávaných kódů odborností (H, F, I, T, S, R, P); (načítá se ze složky *<ciselniky>*) |
| | odbornosti_vyjimky | číselník nezpracovávaných kódů odborností (9H9, 9F9, 2S1); (načítá se ze složky *<ciselniky>*) |
| | odbornosti_delici | číselník kódů dělících odborností (2F1, 2H1); (načítá se ze složky *<ciselniky>*) |
| **Výstupní data** | pripady_temp | soubor sestavených hospitalizačních případů |

| | | |
|---|---|---|
| | doklady_filtr | soubor prvních a posledních dokladů 02 hospitalizačních případů, slouží jako lookup v procesu PGP_VALIDACE_PRIPADU |

Jde o jeden z klíčových procesů, který implementuje metodiku sestavení hospitalizačních případů. Proces zpracovává soubor validních dokladů typu 02 (***pgp_doklady02_k0_validni***), ze kterých sestavuje záznamy hospitalizačních případů dle pravidel metodiky. Záznamy se filtrují dle odborností akutní péče, označují se odbornosti, které mohou vést k rozdělení hospitalizačních případů, a probíhá seřazení dokladů určených *metodikou* (využívá se i číselník ***prijeti_propusteni*** priorit kódů přijetí a propuštění na dokladech 02 pro sestavení případu). Sestavení případů probíhá dle metodikou definovaných pravidel pro spojování hospitalizačních případů. Výsledným případům se přidělí jedinečný identifikátor. Výstup procesu se ukládá do dočasného souboru (***pripady_temp***).

~~*Obrázek 22 Diagram procesu pgp_01_hosp_pripady*~~




Pro sestavení hospitalizačních případů se používá specifické nastavení podmínek v mapě ***hosp_pripady***. Tato mapa prochází postupně jednotlivé doklady 02 seřazené tak, aby následující doklady vždy případně navazovaly na předchozí hospitalizační případ téhož pacienta ve stejném ZZ dle data a dalších třídících parametrů. Údaje z daného předchozího hospitalizačního případu jsou ukládány do proměnných mapy (např. ***poj_id_buf*** s identifikací pojištěnce na předchozím HP).

Při načtení každého dalšího dokladu 02 se hodnoty sledovaných parametrů porovnávají s uloženými hodnotami předchozího případu v proměnných. V případě, kdy údaje z nového dokladu 02 navazují na údaje z předchozího hospitalizačního případu (stejná ZP, IČZ, pojištěnec, nedochází ke změně odbornosti mezi akutní rehabilitační a jiné akutní lůžkové péče, nedošlo k přerušení hospitalizace na více než jeden kalendářní den), jsou údaje z tohoto nového dokladu 02 použity k modifikaci údajů předchozího hospitalizačního případu (datum ukončení, způsob ukončení, propouštěcí odbornost apod.). V opačném případě je předchozí hospitalizační případ poslán na výstup jako sestavený HP a je založen nový hospitalizační případ s údaji z nového dokladu 02.

V rámci tohoto procesu jsou počítány i kalendářní dny v rámci hospitalizačního případu, které pacient stráví celé mimo nemocnici. V případě, kdy je hospitalizační případ sloučen z více dokladů 02 a mezi datem ukončení prvního a datem zahájení druhého je rozdíl dvou dnů, je za každou takovou situaci v hospitalizačním případu připočten jeden den do parametru ***pocet_volnych***. Tyto dny jsou později odpočítány od doby trvání hospitalizačního případu (***los***).

Následující příklad zachycuje podmínku (Java syntaxe), která naplňuje proměnnou ***den_uko_out***, což je identifikace pojištěnce na hospitalizačním případu, který má být již uzavřen a předán na výstup:

```
!input_row.id_poj.equals(Var.id_poj_buf) ||
```

```
input_row.idzz/1000!=Var.idzz_buf/1000 ||
TalendDate.diffDate(input_row.den_zah, Var.den_uko_buf, "dd", true) > 2 ||
(input_row.je_delici != Var.delici_buf) ? Var.den_uko_buf : null
```

Tedy, pokud jsou splněny podmínky pro rozdělení hospitalizačního případu, je proměnná ***den_uko_out*** (předávaná na výstup z mapy) naplněna údajem z proměnné ***Var.den_uko_buf***, kde je uložena hodnota z předchozího hospitalizačního případu. V opačném případě je na výstup předávána hodnota *null*, která je v následujícím kroku vyřazena ze zpracování.

Podobně je proměnná ***den_uko_buf*** v každém kroku naplněna následujícím výrazem:

```
(Var.id_poj_out != null || Relational.ISNULL(Var.den_uko_buf)) ?
input_row.den_uko : (
      TalendDate.compareDate(input_row.den_uko, Var.den_uko_buf) >= 0 ?
      input_row.den_uko : Var.den_uko_buf
)
```

Tedy, pokud je na výstup předáván ukončený HP (má vyplněnu proměnnou ***id_poj_out***) nebo je prázdná proměnná ***den_uko_buf*** (pouze při zpracování úplně prvního hospitalizačního případu), naplní se proměnná ***den_uko_buf*** hodnotou ***den_uko*** z nového záznamu (***input_row***). V opačném případě se do proměnné ***den_uko_buf*** ukládá pozdější z datumů ***den_uko_buf*** a ***den_uko*** nového záznamu.

Proměnné, jejichž hodnota pochází z prvního dokladu 02 (například ***den_zah_buf***) daného případu, jsou plněny následovně:

```
Var.id_poj_buf!=null    &&    Var.id_poj_buf.equals(input_row.id_poj)    &&
Var.id_poj_out==null ? Var.den_zah_buf : input_row.den_zah
```

Tedy nedochází-li k vytvoření nového případu, ponechává se vyplněná hodnota z prvního dokladu.

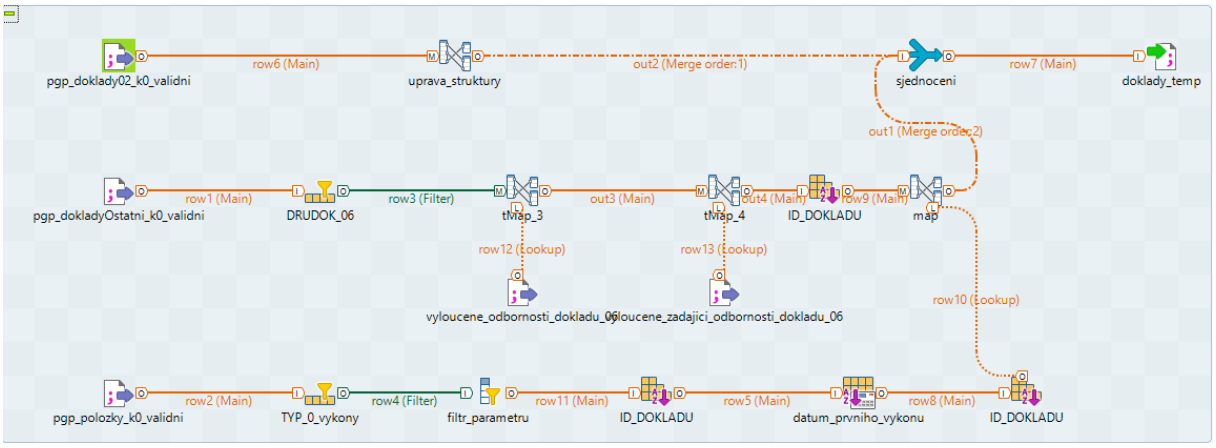
## pgp_02_doklady

*Tabulka 22 Základní popis procesu pgp_02_doklady*

| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~**HTML – Talend dokumentace**~~ | PGP_02_DOKLADY | ~~talend_dokumentace\pgp_02_doklady\pgp_02_doklady_1.4.html~~ |
| **Nadřazený job** | PGP_INICIALNI_PARALELNI | |
| **Předchozí job** | PGP_INICIALNI_PARALELNI | nadřazený job |
| **Následující job** | PGP_VALIDACE_PRIPADU | |
| **Vstupní data** | pgp_doklady02_k0_validni | soubor validních dokladů 02 z procesu PGP_DOKLADY02_VALIDACE_K0 |
| | pgp_dokladyOstatni_k0_validni | soubor validních dokladů 06 z procesu PGP_DOKLADYOSTATNI_VALIDACE_K0 |
| | pgp_polozky_k0_validni | soubor validních dokladů polozky z procesu PGP_POLOZKY_VALIDACE_K0 |
| | vyloucene_odbornosti_dokladu_06 | číselník vyloučených odborností pro doklady typu 06 |
| | vyloucene_zadajici_odbornosti_dokladu_06 | od verze 7.0 nový číselník vyloučených žádajících odborností pro doklady typu 06 |
| **Výstupní data** | doklady_temp | spojený soubor všech dokladů 02 a 06 s dopočteným datem provedení prvního výkonu |

V procesu dochází ke sloučení záznamů dokladů typu 02 a 06. Doklady 06 jsou na vstupu filtrovány pomocí napojení na číselník ***vyloucene_odbornosti_dokladu_06*** a od verze 7.0 nově také ***vyloucene_zadajici_odbornosti_dokladu_06***. K dokladům typu 06 je ze souboru položek dopočítána agregací v komponentě ***datum_prvniho_vykonu*** informace o datu prvního výkonu na tomto dokladu, která bude následně použita k přiřazení celého dokladu k hospitalizačnímu případu. Výstup procesu se ukládá do dočasného souboru ***doklady_temp***.

~~*Obrázek 23 Diagram procesu pgp_02_doklady*~~



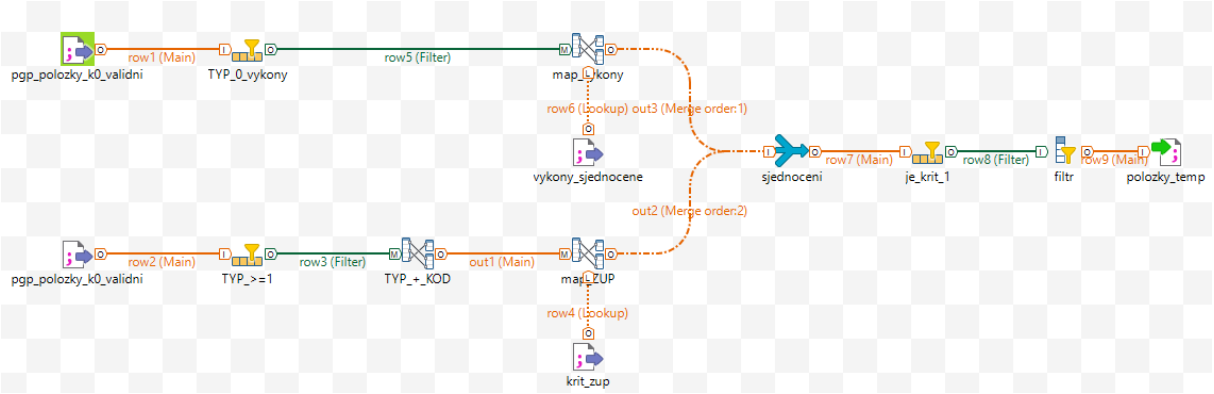

## pgp_03_polozky

*Tabulka 23 Základní popis procesu pgp_03_polozky*

| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~**HTML Talend dokumentace**~~ | PGP_03_POLOZKY | ~~talend_dokumentace\pgp_03_polozky\pgp_03_polozky_1.1.html~~ |
| **Nadřazený job** | PGP_INICIALNI_PARALELNI | |
| **Předchozí job** | PGP_INICIALNI_PARALELNI | nadřazený job |
| **Následující job** | PGP_VALIDACE_PRIPADU | |
| **Vstupní data** | pgp_polozky_k0_validni | soubor validních dokladů 02 z procesu PGP_POLOZKY_VALIDACE_K0 |
| | vykony_sjednocene | seznam všech výkonů, které patří mezi kritické výkony nebo DRG markery, terapeutické výkony, markery UPV, hmotnosti atd. |
| | krit_zup | seznam všech kritických ZUP |
| **Výstupní data** | polozky_temp | spojený soubor sledovaných výkonů a ZUP, přiřazených k dokladu |

V procesu dochází k vyfiltrování vstupních položek na sledované výkony (kritické výkony a DRG markery, terapeutické výkony, výkony UPV, hmotnosti atd.) z číselníku ***vykony_sjednocene*** a zvlášť účtované položky z číselníku ***krit_zup***. Oba typy položek se načítají v samostatných větvích ze souboru ***pgp_polozky_k0_validni*** a následně spojují. Výstup procesu se ukládá do dočasného souboru ***polozky_temp***.

~~*Obrázek 24 Diagram procesu pgp_03_polozky*~~



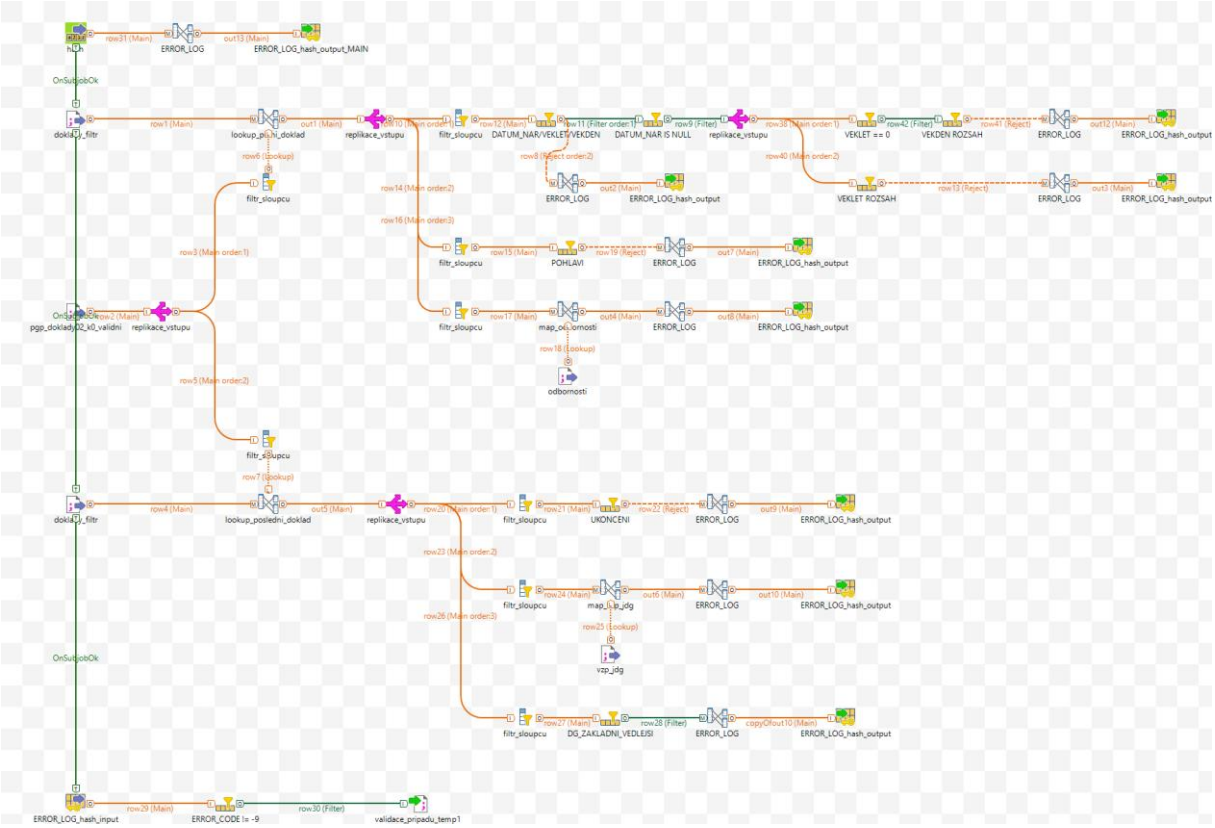

## pgp_validace_pripadu

*Tabulka 24 Základní popis procesu pgp_validace_pripadu*

| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~**Dokumentace**~~ | PGP_VALIDACE_PRIPADU | ~~talend_dokumentace\pgp_validace_pripadu\pgp_validace_pripadu_0.1.html~~ |
| **Nadřazený job** | PGP_INICIALNI_PARALELNI | |
| **Předchozí job** | PGP_01_HOSP_PRIPADY<br>PGP_02_DOKLADY<br>PGP_03_POLOZKY | paralelně spouštěné iniciální procesy |
| **Následující job** | PGP_04A_DOKLADY_PRIPADY_ MASTER | |
| **Vstupní data** | doklady_filtr | soubor prvních a posledních dokladů daného případu z procesu |
| | pgp_doklady02_k0_validni | soubor validních dokladů 02 z procesu PGP_POLOZKY_VALIDACE_K0 |
| **Výstupní data** | validace_pripadu_temp1 | soubor validací na první a poslední doklady 02 hospitalizačního případu |

Proces kontrol prvních a posledních dokladů typu 02 jednotlivých hospitalizačních případů. Kontroluje správně vyplněné údaje pro sestavení případu, jako datum narození nebo věk, pohlaví atd. Informace o nalezených chybách jsou ukládány do dočasného souboru ***validace_pripadu_temp1***.

~~Obrázek 25 Diagram procesu pgp_validace_pripadu~~



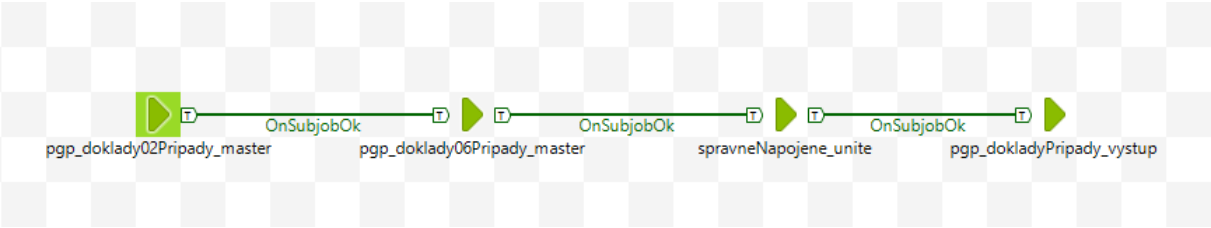
## pgp_04a_doklady_pripady_master

*Tabulka 25 Základní popis procesu pgp_04a_doklady_pripady_master*

| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~HTML - Talend dokumentace~~ | PGP_04A_DOKLADY_PRIPADY_MASTER | ~~talend_dokumentace\pgp_04a_doklady_pripady_master\pgp_04a_doklady_pripady_master_1.1.html~~ |
| **Nadřazený job** | PGP_MASTER | |
| **Předchozí job** | PGP_VALIDACE_PRIPADU | |
| **Následující job** | PGP_DOKLADY02PRIPADY_MASTER | podřízený proces |
| **Vstupní data** | - | |
| **Výstupní data** | - | |

Řídící job napojení dokladů na hospitalizační případy. Tento proces implementuje druhou klíčovou část metodiky sestavení hospitalizačních případů, a to je přiřazení jednotlivých dokladů 02, resp. 06 k hospitalizačním případům, na základě níže definovaných pravidel. Postupně spouští další subjoby napojení dokladů 02 (PGP_DOKLADY02PRIPADY_MASTER), 06 (PGP_DOKLADY06PRIPADY_MASTER) a pomocné subjoby pro vytvoření souboru obsahujícího korektně napojené záznamy (SPRAVNENAPOJENE_UNITE, PGP_DOKLADYPRIPADY_VYSTUP).

~~Obrázek 26 Diagram procesu pgp_04a_doklady_pripady_master~~



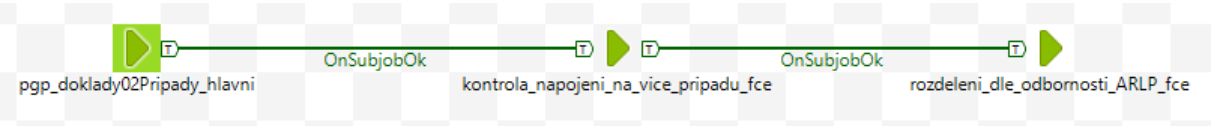

## pgp_doklady02Pripady_master

*Tabulka 26 Základní popis procesu pgp_doklady02Pripady_master*

| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~HTML – Talend dokumentace~~ | PGP_DOKLADY02PRIPADY_MASTER | ~~talend_dokumentace\pgp_04a_doklady_pripady_joby\doklady02\pgp_doklady02Pripady_master\pgp_doklady02Pripady_master_1.0.html~~ |
| Nadřazený job | PGP_04A_DOKLADY_PRIPADY_MASTER | |
| Předchozí job | PGP_04A_DOKLADY_PRIPADY_MASTER | nadřazený proces |
| Následující job | PGP_DOKLADY02PRIPADY_HLAVNI | podřízený proces |
| Vstupní data | - | |
| Výstupní data | - | |

Řídící job napojení dokladů 02 na hospitalizační případy. Spouští postupně podproces provádějící přiřazení dokladu 02 k hospitalizačnímu případu (PGP_DOKLADY02PRIPADY_HLAVNI) a následně dva pomocné procesy, které řeší doklady přiřazené k více možným hospitalizačním případům (KONTROLA_NAPOJENI_NA_VICE_PRIPADU_FCE, ROZDELENI_DLE_ODBORNOSTI_ARLP_FCE).

~~Obrázek 27 Diagram procesu pgp_doklady02Pripady_master~~



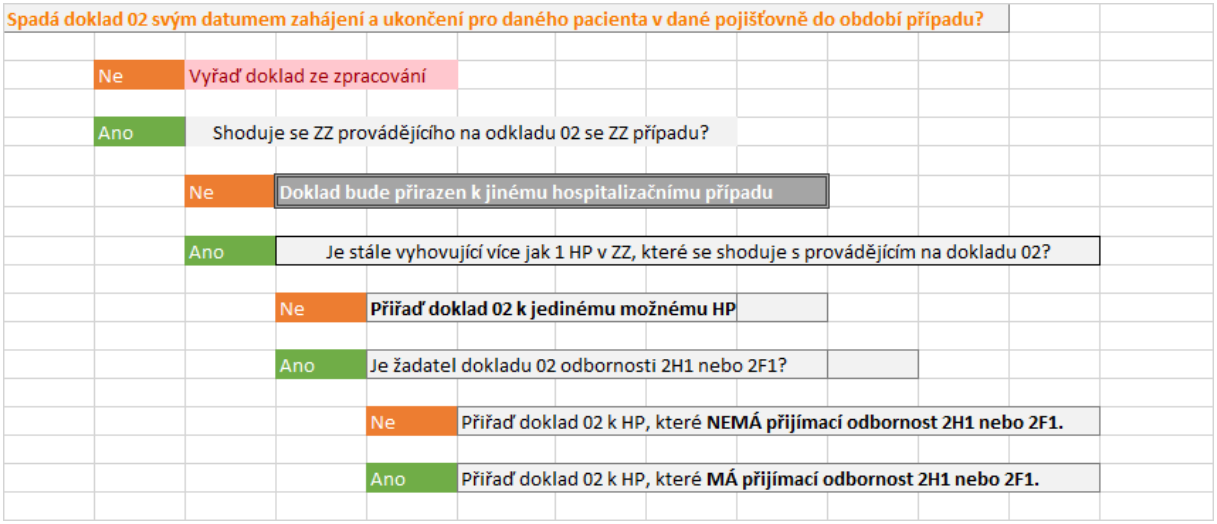
### *pgp_doklady02Pripady_hlavni*

*Tabulka 27 Základní popis procesu pgp_doklady02Pripady_hlavni*

| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~**HTML – Talend dokumentace**~~ | PGP_DOKLADY02PRIPADY_HLAVNI | ~~talend_dokumentace\ pgp_04a_doklady_pripady_joby\ doklady02\ pgp_doklady02Pripady_hlavni\ pgp_doklady02Pripady_hlavni_1.1.html~~ |
| **Nadřazený job** | PGP_DOKLADY02PRIPADY_MASTER | |
| **Předchozí job** | PGP_DOKLADY02PRIPADY_MASTER | nadřazený job |
| **Následující job** | KONTROLA_NAPOJENI_NA_VICE_PRIPADU_FCE | |
| **Vstupní data** | doklady_temp | soubor spojených dokladů 02 a 06 z procesu PGP_02_DOKLADY |
| | pripady_temp | soubor sestavených hospitalizačních případů z procesu PGP_01_HOSP_PRIPADY |
| | odbornosti | číselník prostředních hodnot zpracovávaných kódů odborností (H, F, I, T, S, R, P); (načítá se ze složky *<ciselniky>*) |
| | odbornosti_vyjimky | číselník nezpracovávaných kódů odborností (9H9, 9F9, 2S1); (načítá se ze složky *<ciselniky>*) |
| **Výstupní data** | doklady02_vyrazene | doklady 02 nepřiřazené ke sledovanému hospitalizačnímu případu – budou přiřazeny k jinému HP |
| | doklady02_vicePripadu | doklady 02 přiřazené k jednomu nebo více hospitalizačním případům |
| | doklady02_datum_vyrazene | doklady 02 vyřazené na základě nenapojení přes parametry *DEN_ZAH* (den zahájení) a *DEN_UKO* (den ukončení) |

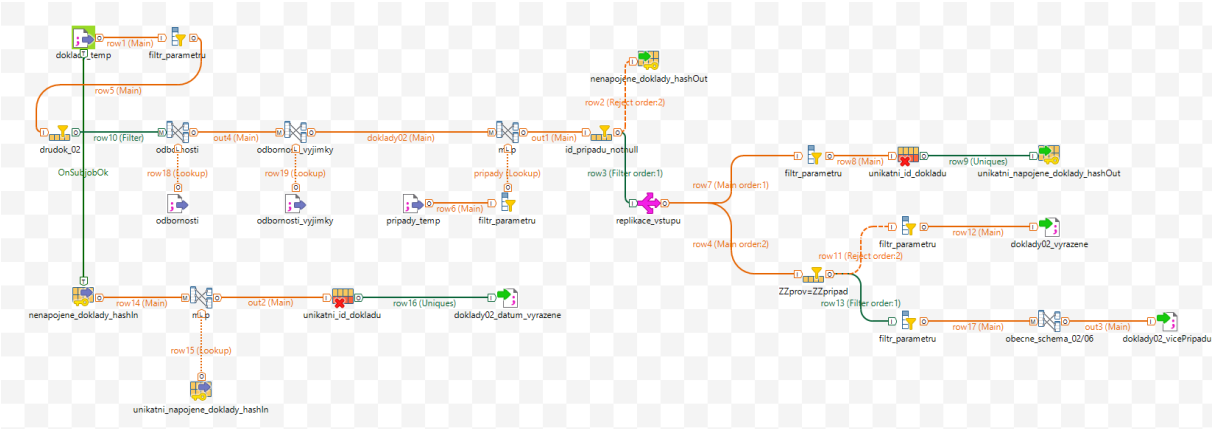
Vyfiltrované doklady 02 ze vstupního souboru ***doklady_temp*** se pomocí klíčových parametrů napojují na soubor sestavených hospitalizačních případů ***pripady_temp***. Prochází se každá dvojice dokladu 02 a hospitalizačního případu, které jsou přiřazeny ke stejnému pojištěnci u stejné ZP, a hledají se doklady, které spadají do období případu. Rozhodovací pravidla jsou zachycena na následujícím obrázku:

*Obrázek 28 Rozhodovací schéma přiřazení dokladů 02 k případům*




Ověřuje se shoda ZZ dokladu 02 se ZZ hospitalizačního případu. Doklady 02, které se v nějakém kroku nenapojily přes datum k vybranému případu, jsou filtrovány přes seznam dokladů, které se podařilo připojit k jinému hospitalizačnímu případu. Na základě těchto pravidel se vytvoří dočasné soubory obsahující vyřazené doklady (***doklady02_datum_vyrazene*** a ***doklady02_vyrazene***) a soubor dokladů přiřazených k hospitalizačním případům (***doklady02_vicePripadu***), které vstupují do podprocesu KONTROLA_NAPOJENI_NA_VICE_PRIPADU_FCE. Oba seznamy vyřazených dokladů by měly být prázdné – všechny doklady 02 sledovaných odborností by měly být přiřazeny k nějakému hospitalizačnímu případu.

~~*Obrázek 29 Diagram procesu pgp_doklady02Pripady_hlavni*~~



*Tabulka 28 Validační soubory vytvořené v rámci procesu*

| Č. | Název | Cesta |
|---|---|---|
| **5** | doklady02_datum_vyrazene | *<tempdir>*/dokladyPripady_temp/ vyrazene/doklady_02_datum_vyrazene.csv |
| **6** | doklady02_vyrazene | *<tempdir>*/dokladyPripady_temp/ vyrazene/doklady02_vyrazene.csv |

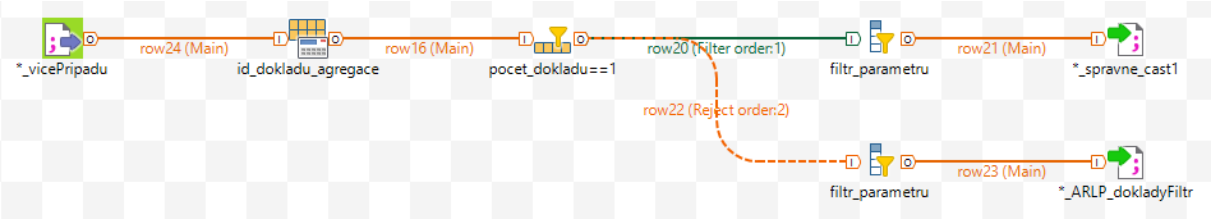
## *kontrola_napojeni_na_vice_pripadu_fce*

*Tabulka 29 Základní popis procesu kontrola_napojeni_na_vice_pripadu_fce*

| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~**HTML – Talend dokumentace**~~ | KONTROLA_NAPOJENI_NA_VICE_PRIPADU_FCE | ~~talend_dokumentace\pgp_04a_doklady_pripady_joby\pomocne_fce\kontrola_napojeni_na_vice_pripadu_fce\kontrola_napojeni_na_vice_pripadu_fce_1.0.html~~ |
| **Nadřazený job** | - | volá se z více procesů |
| **Předchozí job** | - | volá se z více procesů, konkrétní spuštění nastavuje adekvátní kontextové proměnné |
| **Následující job** | ROZDELELNI_DLE_ODBORNOSTI_ARLP_FCE | navazující proces řešící jen vícenásobně přiřazené doklady |
| **Vstupní data** | doklady02_vicePripadu | dle procesu, ze kterého je voláno, konkrétní spuštění nastavuje jméno souboru pomocí adekvátní kontextové proměnné |
| | extramuralni_vicePripadu | |
| | intramuralni_vicePripadu | |
| | nejasnePriraditelna_vicePripadu | |
| **Výstupní data** | doklady02_spravne_cast1 | správně napojené doklady na jeden HP, konkrétní spuštění nastavuje jméno souboru pomocí adekvátní kontextové proměnné |
| | extramuralni_spravne_cast1 | |
| | intramuralni_spravne_cast1 | |
| | nejasnePriraditelna_spravne_cast1 | |
| | doklady02_ARLP_dokladyFiltr | dočasné pomocné soubory - duplicitně napojené doklady na jeden HP, konkrétní spuštění nastavuje jméno souboru pomocí adekvátní kontextové proměnné |
| | extramuralni_ARLP_dokladyFiltr | |
| | intramuralni_ARLP_dokladyFiltr | |
| | nejasnePriraditelna_ARLP_dokladyFiltr | |

Pomocný proces ověřující, zda jsou doklady navázány pouze na jeden hospitalizační případ (agregací přes *id_dokladu* v komponentě ***id_dokladu_agregace***). Volá se z procesů, které přiřazují k hospitalizačním případům doklady 02, resp. 06 pro intramurál, extramurál a nejasně přiřaditelnou péči. V případě napojení pouze jediného hospitalizačního případu jejich seznam ukládá do dočasného souboru ve tvaru ***<typPece>_spravne_cast1*** (typ péče je jeden z „doklady02“, „extramuralni“, „intramuralni“, „nejasnePriraditelna“). Zbylé identifikátory dokladů, napojených na více možných případů, se ukládají do souboru ***<typPece>_ARLP_dokladyFiltr*** a vstupují do dalšího podprocesu ROZDELENI_DLE_ODBORNOSTI_ARLP_FCE.

~~Obrázek 30 Diagram procesu kontrola_napojeni_na_vice_pripadu_fce~~




*Tabulka 30 Validační soubory vytvořené v rámci procesu*

| Č. | Název | Cesta |
|---|---|---|
| **7** | doklady02_spravne_cast1 | *<tempdir>*/dokladyPripady_temp/spravneNapojene/doklady02_doklady02_spravne_cast1.csv |
| **8** | extramuralni_spravne_cast1 | *<tempdir>*/dokladyPripady_temp/spravneNapojene/doklady06_extramuralni_spravne_cast1.csv |
| **9** | intramuralni_spravne_cast1 | *<tempdir>*/dokladyPripady_temp/spravneNapojene/doklady06_intramuralni_spravne_cast1.csv |
| **10** | nejasnePriraditelna_spravne_cast1 | *<tempdir>*/dokladyPripady_temp/spravneNapojene/doklady06_nejasnePriraditelna_spravne_cast1.csv |

## *rozdeleni_dle_odbornosti_ARLP_fce*

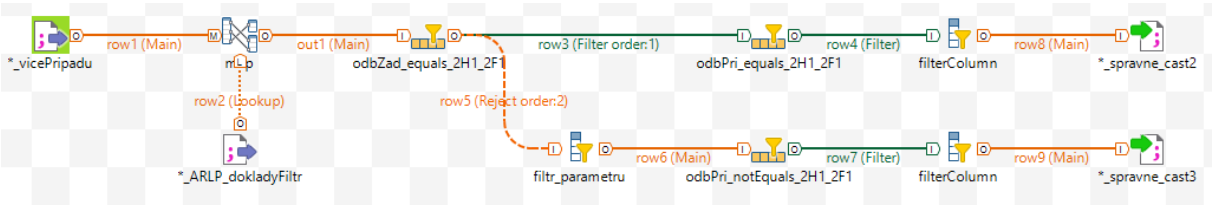
*Tabulka 31 Základní popis procesu rozdeleni_dle_odbornosti_ARLP_fce*

| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~**HTML – Talend dokumentace**~~ | ROZDELENI_DLE_ODBORNOSTI_ARLP_FCE | ~~talend_dokumentace\pgp_04a_doklady_pripady_joby\pomocne_fce\rozdeleni_dle_odbornosti_ARLP_fce\rozdeleni_dle_odbornosti_ARLP_fce_1.0.html~~ |
| **Nadřazený job** | - | volá se z více procesů |
| **Předchozí job** | KONTROLA_NAPOJENI_NA_VICE_PRIPADU_FCE | volá se z více procesů, konkrétní spuštění nastavuje adekvátní kontextové proměnné |
| **Následující job** | - | dle kontextu volání |
| **Vstupní data** | doklady02_vicePripadu | dle procesu, ze kterého je voláno, konkrétní spuštění nastavuje jméno souboru pomocí adekvátní kontextové proměnné; soubory ***_ARLP_dokladyFiltr*** byly vytvořeny v předchozím procesu KONTROLA_NAPOJENI_NA_VICE_PRIPADU_FCE |
| | extramuralni_vicePripadu | |
| | intramuralni_vicePripadu | |
| | nejasnePriraditelna_vicePripadu | |
| | doklady02_ARLP_dokladyFiltr | |
| | extramuralni_ARLP_dokladyFiltr | |
| | intramuralni_ARLP_dokladyFiltr | |
| | nejasnePriraditelna_ARLP_dokladyFiltr | |
| **Výstupní data** | doklady02_spravne_cast2 | správně napojené doklady na HP akutní lůžkové rehabilitace |
| | extramuralni_spravne_cast2 | |
| | intramuralni_spravne_cast2 | |
| | nejasnePriraditelna_spravne_cast2 | |

| | | |
|---|---|---|
| | doklady02_spravne_cast3 | správně napojené doklady na HP jiné akutní lůžkové péče |
| | extramuralni_spravne_cast3 | |
| | intramuralni_spravne_cast3 | |
| | nejasnePriraditelna_spravne_cast3 | |

Druhá pomocná funkce, která provádí finální rozdělení dokladů napojených na více možných případů. Rozdělení probíhá na základě logiky vykázání (resp. nevykázání) dokladu v odbornosti akutní rehabilitační lůžkové péče a srovnání s příjmovou odborností na kontrolovaném hospitalizačním případu.

~~*Obrázek 31 Diagram procesu rozdeleni_dle_odbornosti_ARLP_fce*~~




Doklady, které byly finálně přiřazeny k jedinému hospitalizačnímu případu, jsou uloženy do souborů ***<typPece>_spravne_cast2*** (pro odbornost akutní rehabilitační lůžkové péče) a ***<typPece>_spravne_cast3*** (pro ostatní odborností akutních lůžek).

*Tabulka 32 Validační soubory vytvořené v rámci procesu*

| Č. | Název | Cesta |
|---|---|---|
| **11** | doklady02_spravne_cast2 | *<tempdir>*/dokladyPripady_temp/spravneNapojene/doklady02_doklady02_spravne_cast2.csv |
| **12** | extramuralni_spravne_cast2 | *<tempdir>*/dokladyPripady_temp/spravneNapojene/doklady06_extramuralni_spravne_cast2.csv |
| **13** | intramuralni_spravne_cast2 | *<tempdir>*/dokladyPripady_temp/spravneNapojene/doklady06_intramuralni_spravne_cast2.csv |
| **14** | nejasnePriraditelna_spravne_cast2 | *<tempdir>*/dokladyPripady_temp/spravneNapojene/doklady06_nejasnePriraditelna_spravne_cast2.csv |
| **15** | doklady02_spravne_cast3 | *<tempdir>*/dokladyPripady_temp/spravneNapojene/doklady02_doklady02_spravne_cast3.csv |
| **16** | extramuralni_spravne_cast3 | *<tempdir>*/dokladyPripady_temp/spravneNapojene/doklady06_extramuralni_spravne_cast3.csv |
| **17** | intramuralni_spravne_cast3 | *<tempdir>*/dokladyPripady_temp/spravneNapojene/doklady06_intramuralni_spravne_cast3.csv |
| **18** | nejasnePriraditelna_spravne_cast3 | *<tempdir>*/dokladyPripady_temp/spravneNapojene/doklady06_nejasnePriraditelna_spravne_cast3.csv |

## pgp_doklady06Pripady_master

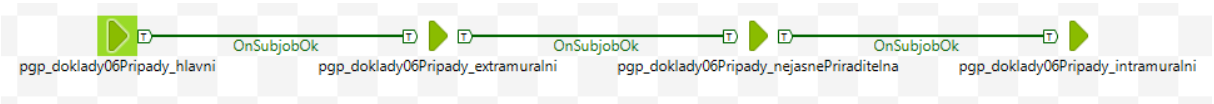
*Tabulka 33 Základní popis procesu pgp_doklady06Pripady_master*

| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~**HTML – Talend dokumentace**~~ | PGP_DOKLADY06PRIPADY_MASTER | ~~talend_dokumentace\pgp_04a_doklady_pripady_joby\doklady06\pgp_doklady06Pripady_master\~~ |

| | | |
|---|---|---|
| | | ~~pgp_doklady06Pripady_master_1.1.html~~ |
| **Nadřazený job** | PGP_04A_DOKLADY_PRIPADY_ MASTER | |
| **Předchozí job** | PGP_DOKLADY02PRIPADY_MASTER | |
| **Následující job** | PGP_DOKLADY06PRIPADY_HLAVNI | podřízený proces |
| **Vstupní data** | - | |
| **Výstupní data** | - | |

Řídící job napojení dokladů 06 na hospitalizační případy. Spouští nejprve hlavní podproces pro přiřazení dokladů k případům (PGP_DOKLADY06PRIPADY_HLAVNI) a dále podprocesy, které řeší jednotlivé dílčí typy péče (PGP_DOKLADY06PRIPADY_EXTRAMURALNI, PGP_DOKLADY06PRIPADY_INTRAMURALNI, PGP_DOKLADY06PRIPADY_NEJASNEPRIRADITELNA).

~~*Obrázek 32 Diagram procesu pgp_doklady06Pripady_master*~~



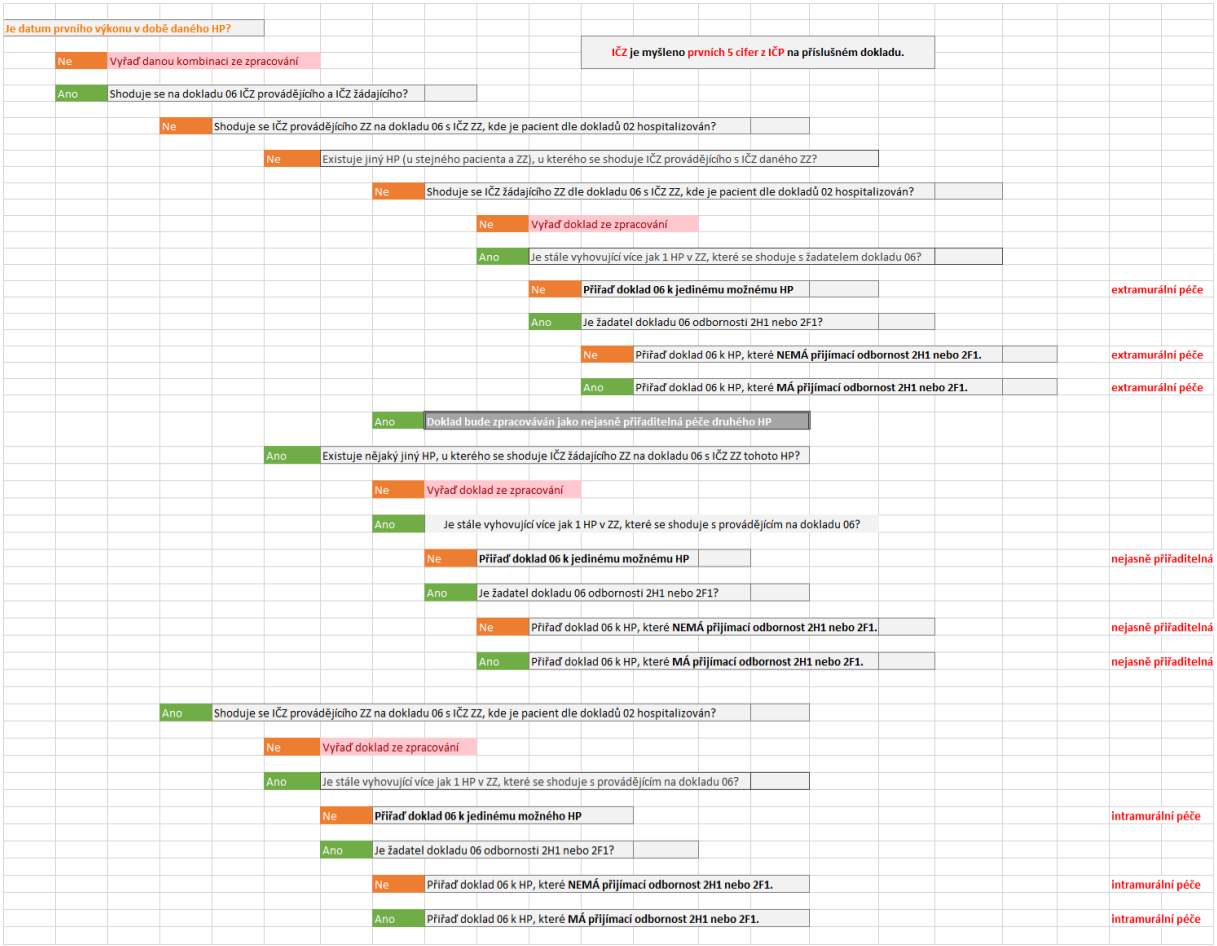

## *pgp_doklady06Pripady_hlavni*

*Tabulka 34 Základní popis procesu pgp_doklady06Pripady_hlavni*

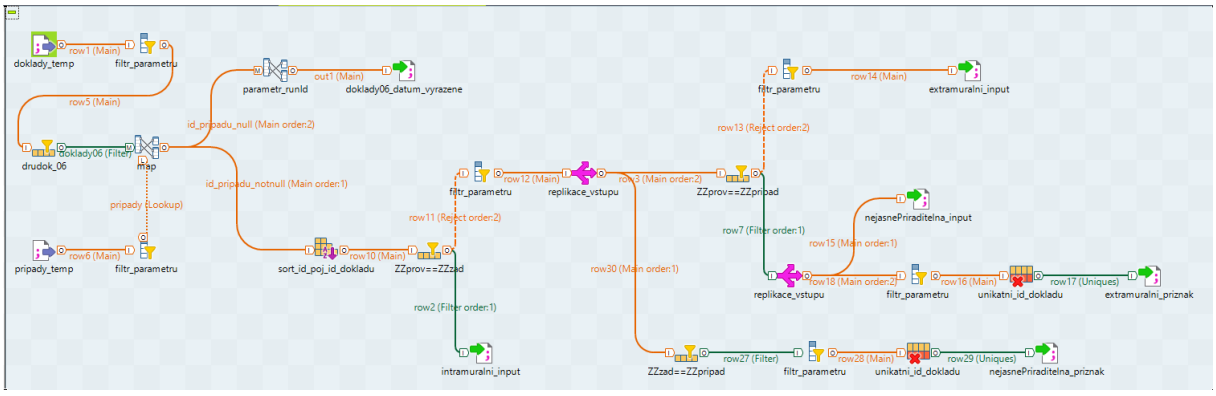
| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~**HTML – Talend dokumentace**~~ | PGP_DOKLADY06PRIPADY_HLAVNI | ~~talend_dokumentace\pgp_04a_doklady_pripady_joby\doklady06\pgp_doklady06Pripady_hlavni\pgp_doklady06Pripady_hlavni_1.1.html~~ |
| **Nadřazený job** | PGP_DOKLADY06PRIPADY_MASTER | |
| **Předchozí job** | PGP_DOKLADY06PRIPADY_MASTER | nadřazený job |
| **Následující job** | PGP_DOKLADY06PRIPADY_EXTRAMURALNI | |
| **Vstupní data** | doklady_temp | soubor spojených dokladů 02 a 06 z procesu PGP_02_DOKLADY |
| | pripady_temp | soubor sestavených hospitalizačních případů z procesu |
| **Výstupní data** | extramural_input | doklady 06 potenciálně přiřaditelné k HP jako extramurální péče |
| | intramural_input | doklady 06 potenciálně přiřaditelné k HP jako intramurální péče |
| | nejasnePriraditelna_input | doklady 06 potenciálně přiřaditelné k HP jako nejasně přiřaditelná péče |
| | extramural_priznak | ID dokladů, které mohou být přiřazeny k HP jako extramurální/nejasně přiřaditelná péče |
| | nejasnePriraditelna_priznak | ID dokladů 06, vyžádané v rámci jednoho ZZ hospitalizujícího pacienta a provedeny v jiném ZZ |
| | doklady06_datum_vyrazene | doklady 06 vyřazené na základě nenapojení přes parametry *DEN_ZAH* (den zahájení) a *DEN_UKO* (den ukončení) |

Vyfiltrované doklady 06 ze vstupního souboru ***doklady_temp*** jsou pomocí klíčových parametrů napojeny na soubor sestavených hospitalizačních případů ***pripady_temp***. Prochází se každá dvojice hospitalizačního případu a dokladu 06 u stejného pojištěnce a zdravotní pojišťovny. U dokladu 06 je načítán i dopočtený parametr *datum_prvniho_vykonu* a ověřuje se příslušnost tohoto data do období trvání hospitalizačního případu (v komponentě ***map***). Dle metodiky přiřazení definované rozhodovacím stromem jsou doklady rozděleny do tří souborů podle typu lůžkové péče (***intramural_input***, ***extramural_input*** a ***nejasnePriraditelna_input***) ve vztahu ke sledovaným hospitalizačním případům, ke kterým mohou být doklady přiřazeny. Tyto výstupní soubory se dále zpracovávají v samostatných podprocesech. Rozhodovací pravidla jsou zobrazena na následujícím obrázku:

*Obrázek 33 Rozhodovací schéma přiřazení dokladů 06 k hospitalizačním případům*



~~*Obrázek 34 Diagram procesu pgp_doklady06Pripady_hlavni*~~




Kromě výše popsaných tří výstupních souborů s přiřazením typu péče k hospitalizačním dokladům jsou vytvořeny dva doplňující soubory identifikaci dokladů, které mohou hrát roli v následujících podprocesech. Za prvé je to soubor ***nejasnePriraditelna_priznak***, který zahrnuje doklady, které byly vyžádány v rámci ZZ, kde byl pacient hospitalizován, ale byly provedeny v jiném ZZ. Tento soubor se pak použije v procesu PGP_DOKLADY06PRIPADY_NEJASNEPRIRADITELNA, kde se dohledává péče vyžádaná v průběhu jedné hospitalizace v jednom ZZ a provedena v druhém ZZ, kde je následně pacient také hospitalizován. Druhý soubor ***extramuralni_priznak*** obsahuje seznam dokladů, které mohou být přiřazeny jako intramurální péče k nějakému případu. Tento soubor se pak použije v procesu PGP_DOKLADY06PRIPADY_EXTRAMURALNI k odfiltrování dokladů, které by mohly být zároveň přiřazeny jako extramurální péče k jinému hospitalizačnímu případu.

Vyřazené doklady, které nebyly přiřazeny k žádnému případu přes parametr datum prvního výkonu, jsou uloženy do samostatného dočasného souboru **doklady06_datum_vyrazene**. Do tohoto souboru byla pro větší názornost doplněna položka id_pripadu a také nepovinná položka RUN_ID v závislosti na použití parametru runId.

*Tabulka 35 Validační soubory vytvořené v rámci procesu*

| Č. | Název | Cesta |
|---|---|---|
| **19** | doklady06_datum_vyrazene | *<tempdir>*/dokladyPripady_temp/ vyrazene/doklady_06_datum_vyrazene.csv |

## *pgp_doklady06Pripady_extramuralni*

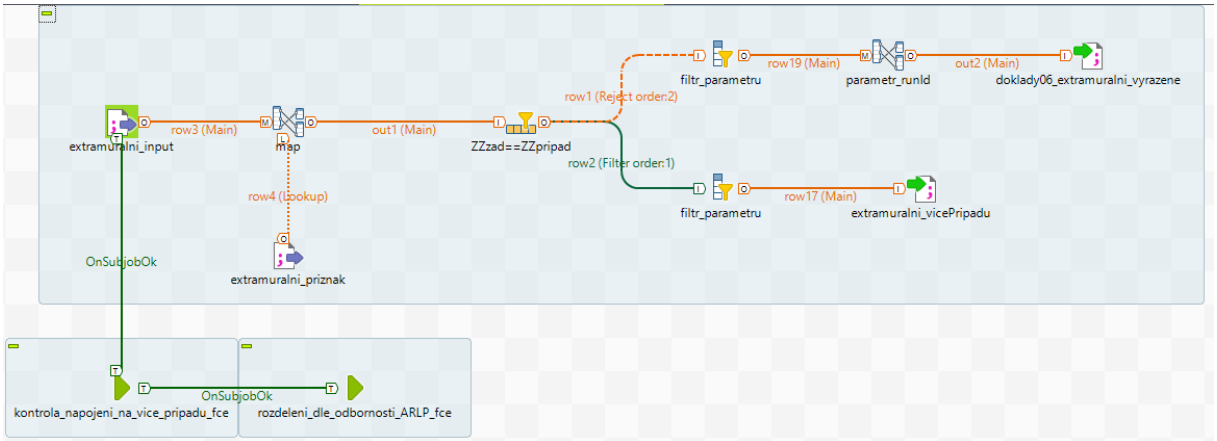
*Tabulka 36 Základní popis procesu pgp_doklady06Pripady_extramuralni*

| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~**HTML – Talend dokumentace**~~ | PGP_DOKLADY06PRIPADY_ EXTRAMURALNI | ~~talend_dokumentace\ pgp_04a_doklady_pripady_joby\doklady06\ pgp_doklady06Pripady_extramuralni\ pgp_doklady06Pripady_extramuralni_1.1.html~~ |
| **Nadřazený job** | PGP_DOKLADY06PRIPADY_MASTER | |
| **Předchozí job** | PGP_DOKLADY06PRIPADY_HLAVNI | nadřazený job |
| **Následující job** | PGP_DOKLADY06PRIPADY_INTRAMURALNI | |

| | | |
|---|---|---|
| **Vstupní data** | extramuralni_input | soubor dokladů 06 z procesu PGP_DOKLADY06PRIPADY_HLAVNI, přiřazených k hospitalizačnímu případu, kde se shoduje ZZ případu se ZZ žadatele na dokladu |
| | extramuralni_priznak | ID dokladů 06 z procesu PGP_DOKLADY06PRIPADY_HLAVNI, které lze přiřadit k nějakému hospitalizačnímu případu přes ZZ provádějícího na dokladu |
| **Výstupní data** | doklady06_extramuralni_vyrazene | doklady 06 nepřiřazené ke sledovanému hospitalizačnímu případu |
| | extramuralni_vicePripadu | dočasný pomocný soubor - doklady 06 přiřazené k jednomu nebo více hospitalizačním případům jako extramurální péče |

Dle pravidel definovaných v rozhodovacím stromu pro napojení dokladů v případě extramurální péče jsou vytvořeny výstupní soubory správně a nesprávně napojených dokladů. Vybírány jsou doklady ze souboru ***extramuralni_input***, u kterých se ZZ žadatele shoduje se ZZ nemocnice, kde byl pacient hospitalizován. Zároveň jsou vyřazeny ty kombinace, u kterých lze doklad 06 přiřadit k jinému hospitalizačnímu případu jako nejasně přiřaditelná péče (pomocí souboru ***extramuralni_priznak***).

~~*Obrázek 35 Diagram procesu pgp_doklady06Pripady_extramuralni*~~




Napojené doklady 06 jsou uloženy do souboru ***extramuralni_vicePripadu*** a dále zpracovány v pomocných funkcích KONTROLA_NAPOJENI_NA_VICE_PRIPADU_FCE a ROZDELENI_DLE_ODBORNOSTI_ARLP_FCE. Význam a popis zmíněných funkcí viz výše. Doklady nenapojené na žádný hospitalizační případ jsou uloženy do souboru ***doklady06_extramuralni_vyrazene***. Do tohoto souboru se také ukládá nepovinná položka RUN_ID v závislosti na použití parametru runId.

*Tabulka 37 Validační soubory vytvořené v rámci procesu*

| Č. | Název | Cesta |
|---|---|---|

| | | |
|---|---|---|
| **20** | doklady06_extramuralni_vyrazene | *<tempdir>*/dokladyPripady_temp/ doklady06_extramuralni_vyrazene.csv |

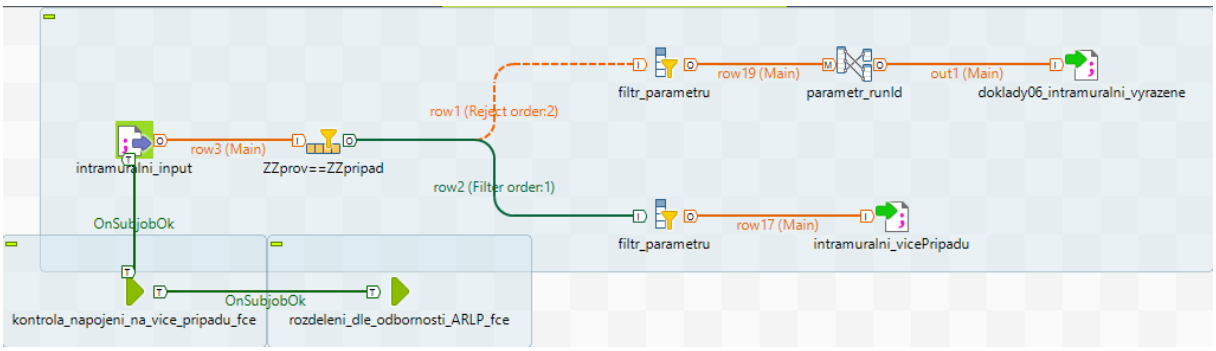
### *pgp_doklady06Pripady_intramuralni*

*Tabulka 38 Základní popis procesu pgp_doklady06Pripady_intramuralni*

| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~**HTML – Talend dokumentace**~~ | PGP_DOKLADY06PRIPADY_ INTRAMURALNI | ~~talend_dokumentace\ pgp_04a_doklady_pripady_joby\doklady06\ pgp_doklady06Pripady_intramuralni\ pgp_doklady06Pripady_intramuralni_1.1.html~~ |
| **Nadřazený job** | PGP_DOKLADY06PRIPADY_MASTER | |
| **Předchozí job** | PGP_DOKLADY06PRIPADY_EXTRAMURALNI | nadřazený job |
| **Následující job** | PGP_DOKLADY06PRIPADY_ NEJASNEPRIRADITELNA | |
| **Vstupní data** | intramuralni_input | soubor dokladů 06 z procesu PGP_DOKLADY06PRIPADY_HLAVNI, přiřazených k hospitalizačnímu případu, kde se shoduje ZZ provádějícího pracoviště se ZZ žadatele na dokladu |
| **Výstupní data** | doklady06_intramuralni_vyrazene | doklady 06 nepřiřazené ke sledovanému hospitalizačnímu případu |
| | intramuralni_vicePripadu | dočasný pomocný soubor - doklady 06 přiřazené k jednomu nebo více hospitalizačním případům jako intramurální péče |

Dle pravidel definovaných v rozhodovacím stromu pro napojení dokladů v případě intramurální péče jsou vytvořeny výstupní soubory napojených a nenapojených dokladů. Jako intramurální péče jsou přiřazeny pouze doklady ze souboru ***intramuralni_input***, u kterých se ZZ provádějícího pracoviště (a v tomto případě i žadatele) shoduje se ZZ nemocnice, kde byl pacient hospitalizován.

~~*Obrázek 36 Diagram procesu pgp_doklady06Pripady_intramuralni*~~




Korektně napojené doklady 06 jsou uloženy do souboru ***intramuralni_vicePripadu*** a dále zpracovány v pomocných funkcích KONTROLA_NAPOJENI_NA_VICE_PRIPADU_FCE a ROZDELENI_DLE_ODBORNOSTI_ARLP_FCE. Význam a popis zmíněných funkcí viz výše. Doklady

nenapojené na žádný hospitalizační případ jsou uloženy do souboru ***doklady06_intramuralni_vyrazene***. Do tohoto souboru se také ukládá nepovinná položka RUN_ID v závislosti na použití parametru runId.

*Tabulka 39 Validační soubory vytvořené v rámci procesu*

| Č. | Název | Cesta |
|---|---|---|
| **21** | doklady06_intramuralni_vyrazene | *<tempdir>*/dokladyPripady_temp/ doklady06_intramuralni_vyrazene.csv |

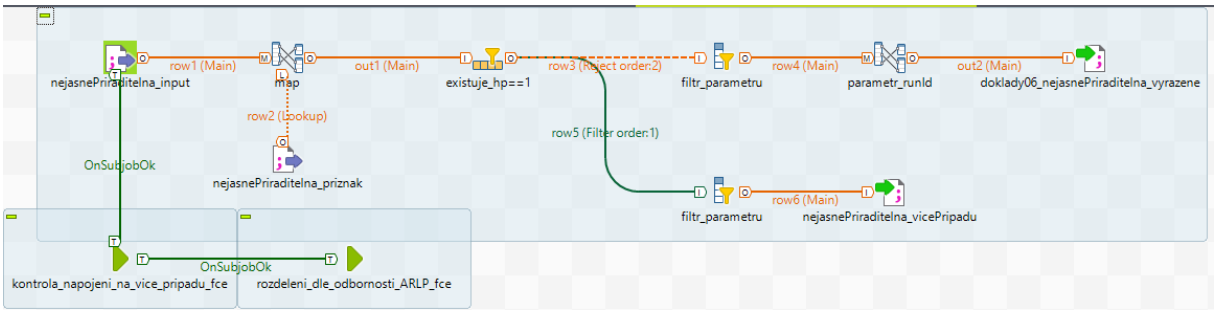
## *pgp_doklady06Pripady_nejasnePriraditelna*

*Tabulka 40 Základní popis procesu pgp_doklady06Pripady_nejasnePriraditelna*

| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~**HTML – Talend dokumentace**~~ | PGP_DOKLADY06PRIPADY_ NEJASNEPRIRADITELNA | ~~talend_dokumentace\ pgp_04a_doklady_pripady_joby\doklady06\ pgp_doklady06Pripady_nejasnePriraditelna\ pgp_doklady06Pripady_nejasnePriraditelna_1.1.html~~ |
| **Nadřazený job** | PGP_DOKLADY06PRIPADY_MASTER | |
| **Předchozí job** | PGP_DOKLADY06PRIPADY_INTRAMURALNI | nadřazený job |
| **Následující job** | KONTROLA_NAPOJENI_NA_VICE_PRIPADU_FCE | |
| **Vstupní data** | nejasnePriraditelna_input | soubor dokladů 06 z procesu PGP_DOKLADY06PRIPADY_HLAVNI přiřazených k hospitalizačnímu případu, kde se shoduje ZZ případu se ZZ provádějícího na tomto dokladu |
| | nejasnePriraditelna_priznak | ID dokladů 06 z procesu PGP_DOKLADY06PRIPADY_HLAVNI, které lze přiřadit k nějakému hospitalizačnímu případu přes ZZ žádajícího na dokladu |
| **Výstupní data** | doklady06_nejasnePriraditelna_ vyrazene | doklady 06 nepřiřazené ke sledovanému hospitalizačnímu případu |
| | nejasnePriraditelna_vicePripadu | dočasný pomocný soubor - doklady 06 přiřazené k jednomu nebo více hospitalizačním případům jako nejasně přiřaditelná péče |

Dle pravidel definovaných v rozhodovacím stromu pro napojení dokladů v případě nejasně přiřaditelné péče jsou vytvořeny výstupní soubory napojených a nenapojených dokladů. Vybírány jsou doklady 06 ze souboru ***nejasnePriraditelna_input***, u kterých se ZZ provádějícího shoduje se ZZ nemocnice, kde byl pacient hospitalizován. Zároveň jsou vybrány (v komponentě ***map***) pouze ty doklady 06 ze souboru ***nejasnePriraditelna_priznak***, které byly vyžádány v průběhu jiného hospitalizačnímu případu (ZZ žadatele dokladu 06 se shoduje se ZZ hospitalizujícího pacienta).

~~*Obrázek 37 Diagram procesu pgp_doklady06Pripady_nejasnePriraditelna*~~




Korektně napojené doklady 06 jsou uloženy do souboru **nejasnePriraditelna_vicePripadu** a dále zpracovány v pomocných funkcích KONTROLA_NAPOJENI_NA_VICE_PRIPADU_FCE a ROZDELENI_DLE_ODBORNOSTI_ARLP_FCE. Význam a popis zmíněných funkcí viz výše. Doklady nenapojené na žádný hospitalizační případ jsou uloženy do souboru **doklady06_nejasnePriraditelna_vyrazene**. Do tohoto souboru se také ukládá nepovinná položka RUN_ID v závislosti na použití parametru runId.

*Tabulka 41 Validační soubory vytvořené v rámci procesu*

| Č. | Název | Cesta |
|---|---|---|
| **22** | doklady06_nejasnePriraditelna_vyrazene | *<tempdir>*/dokladyPripady_temp/ doklady06_nejasnePriraditelna_vyrazene.csv |

## spravneNapojene_unite

*Tabulka 42 Základní popis procesu spravneNapojene_unite*

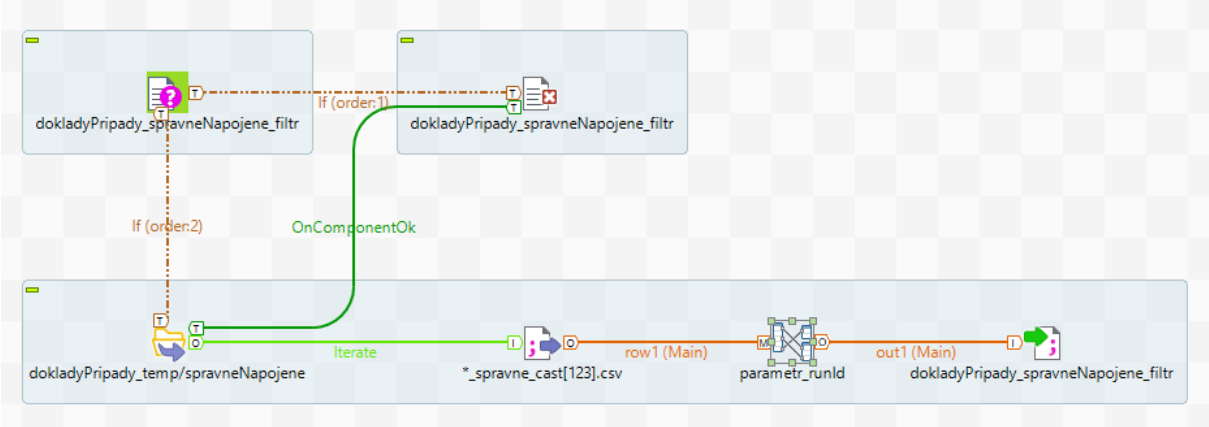
| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~**HTML – Talend dokumentace**~~ | SPRAVNENAPOJENE_UNITE | ~~talend_dokumentace\ pgp_04a_doklady_pripady_joby\ spravneNapojene_unite\ spravneNapojene_unite_1.2.html~~ |
| **Nadřazený job** | PGP_04A_DOKLADY_PRIPADY_MASTER | |
| **Předchozí job** | PGP_DOKLADY06PRIPADY_NEJASNEPRIRADITELNA | |
| **Následující job** | PGP_DOKLADYPRIPADY_VYSTUP | |
| **Vstupní data** | doklady02_spravne_cast1 | korektně napojené doklady jednotlivých typů k hospitalizačním případům z procesů PGP_DOKLADY02PRIPADY_MASTER a PGP_DOKLADY06PRIPADY_MASTER |
| | extramuralni_spravne_cast1 | |
| | intramuralni_spravne_cast1 | |
| | nejasnePriraditelna_spravne_cast1 | |
| | doklady02_spravne_cast2 | |
| | extramuralni_spravne_cast2 | |
| | intramuralni_spravne_cast2 | |
| | nejasnePriraditelna_spravne_cast2 | |
| | doklady02_spravne_cast3 | |
| | extramuralni_spravne_cast3 | |
| | intramuralni_spravne_cast3 | |
| | nejasnePriraditelna_spravne_cast3 | |

| | | |
|---|---|---|
| **Výstupní data** | dokladyPripady_spravneNapojene_filtr | správně napojené doklady na HP všech typů dohromady |

Pomocný proces prochází v iteračních krocích soubory uložené v složce napojených dokladů (*<tempdir>*/dokladyPripady_temp/spravneNapojene/) a skládá je do jednoho společného souboru ***dokladyPripady_spravneNapojene_filtr***. Do tohoto souboru se také ukládá nepovinná položka RUN_ID v závislosti na použití parametru runId.

V případě, že soubor existuje, je před spuštěním procesu z tempové složky vymazán.

~~*Obrázek 38 Diagram procesu spravneNapojene_unite*~~




## pgp_dokladyPripady_vystup

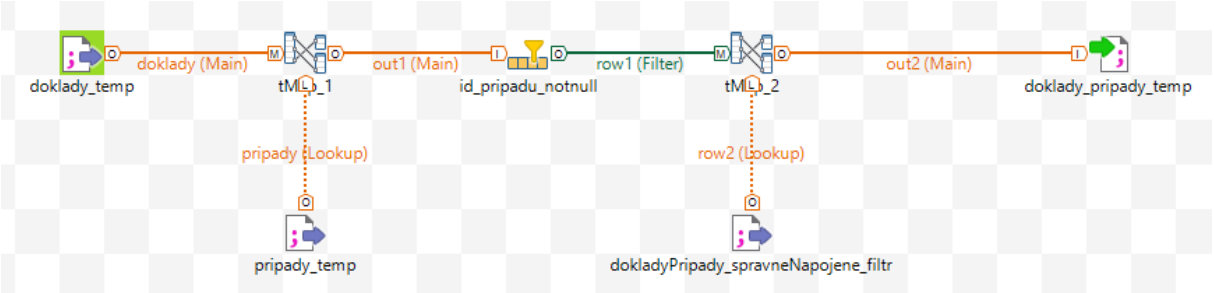
*Tabulka 43 Základní popis procesu pgp_dokladyPripady_vystup*

| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~**HTML Talend dokumentace**~~ | PGP_DOKLADYPRIPADY_VYSTUP | ~~talend_dokumentace\ pgp_04a_doklady_pripady_joby\ pgp_dokladyPripady_vystup\ pgp_dokladyPripady_vystup_1.0.html~~ |
| **Nadřazený job** | PGP_04A_DOKLADY_PRIPADY_MASTER | |
| **Předchozí job** | SPRAVNENAPOJENE_UNITE | |
| **Následující job** | PGP_04B_POLOZKY_CISELNIKY | |
| **Vstupní data** | doklady_temp | soubor spojených dokladů 02 a 06 z procesu PGP_02_DOKLADY |
| | pripady_temp | soubor sestavených hospitalizačních případů z procesu PGP_01_HOSP_PRIPADY |
| | dokladyPripady_spravneNapojene_filtr | soubor kombinací správně napojených dokladů na hospitalizační případy z procesu SPRAVNENAPOJENE_UNITE |

| | | |
|---|---|---|
| **Výstupní data** | doklady_pripady_temp | spojený soubor všech dokladů 02 a 06 s korektně napojeným hospitalizačním případem |

Proces opětovně napojuje doklady obou typů na soubor složených hospitalizačních případů přes klíčové parametry. Ze všech dvojic jsou vybrány pouze korektní přiřazení dokladu na hospitalizační případ ze souboru ***dokladyPripady_spravneNapojene_filtr***, složeného v předchozím kroku. Výstup procesu se ukládá do dočasného souboru ***doklady_pripady_temp***.

~~*Obrázek 39 Diagram procesu pgp_dokladyPripady_vystup*~~



## pgp_04b_polozky_ciselniky

*Tabulka 44 Základní popis procesu pgp_04b_polozky_ciselniky*

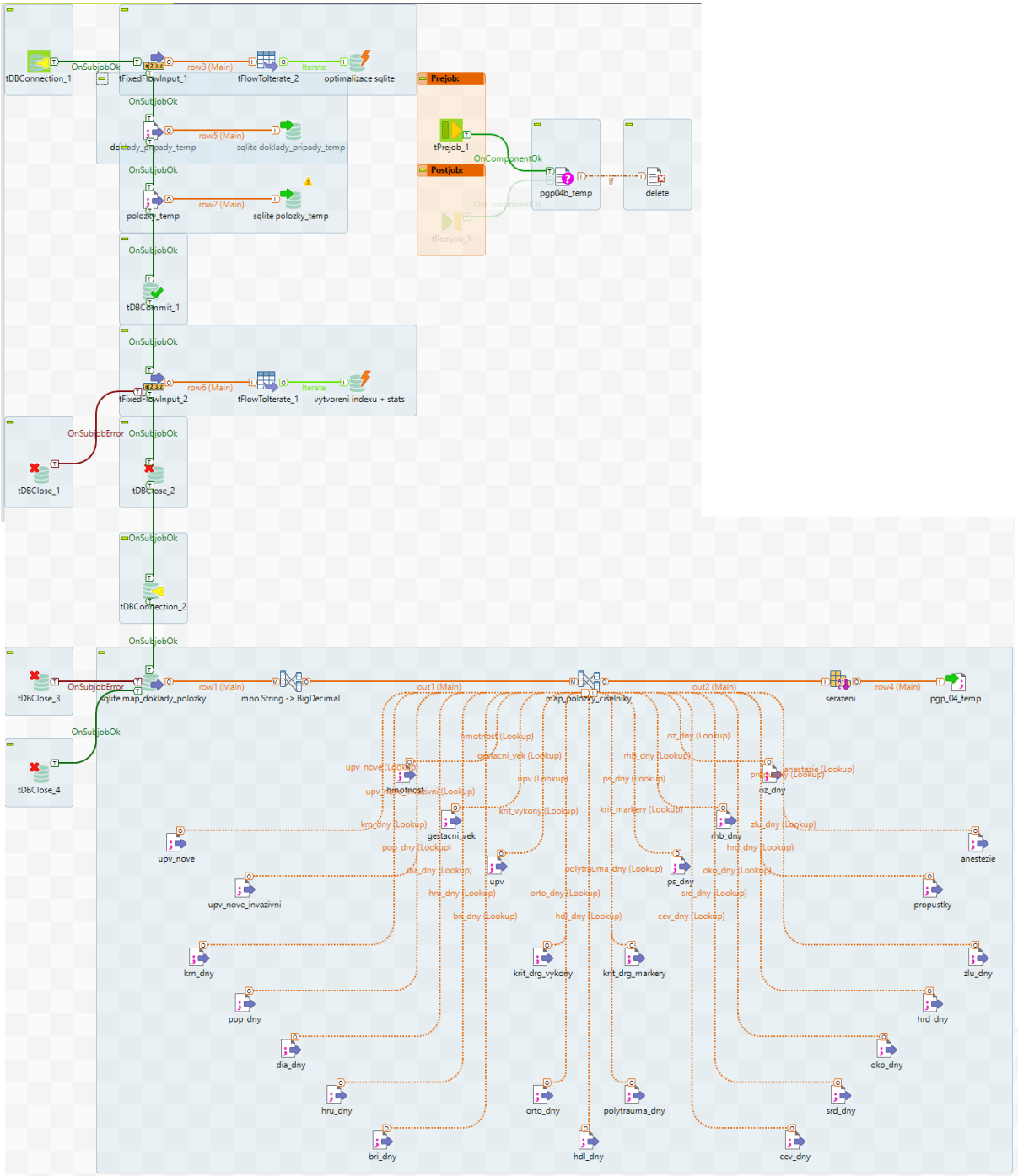
| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~HTML Talend dokumentace~~ | PGP_04B_POLOZKY_CISELNIKY | ~~talend_dokumentace\pgp_04b_polozky_ciselniky\pgp_04b_polozky_ciselniky_2.0.html~~ |
| **Nadřazený job** | PGP_MASTER | |
| **Předchozí job** | PGP_DOKLADYPRIPADY_VYSTUP | |
| **Následující job** | PGP_05A_AGREGACE_POLOZEK | |
| **Vstupní data** | doklady_pripady_temp | doklady korektně napojené na hospitalizační případy v kroku PGP_DOKLADYPRIPADY_VYSTUP |
| | polozky_temp | položky přiřazené k dokladům, které jsou u pacientů s hospitalizačními případy, vzniklé v procesu PGP_03_POLOZKY |
| | krit_drg_vykony | číselník kritických výkonů DRG |
| | krit_drg_markery | číselník DRG markerů |
| | hmotnost | číselník výkonů pro porodní hmotnost |
| | gestacni_vek | číselník výkonů pro gestační věk novorozence |
| | upv | číselník markerů pro umělou plicní ventilaci převzatých z IR-DRG (ve dnech včetně rozmezí) |
| | upv_nove | číselník markerů pro délku umělé plicní ventilace (invazivní, neinvazivní, HFNO) |
| | upv_nove_invazivni | číselník markerů pro délku invazivní a neinvazivní umělé plicní ventilace |
| | propustky | číselník výkonů pro OD propustek |
| | anestezie | číselník výkonů anestezie |
| | oz_dny | číselník výkonů ozařovacích dnů |
| | rhb_dny | číselník výkonů rehabilitačních dnů |
| | ps_dny | číselník výkonů dnů akutní psychiatrické péče |
| | krn_dny | číselník výkonů dnů s operačním výkonem na lebce nebo mozku |
| | pop_dny | číselník výkonů ošetřovacích dnů pro popáleninu nebo omrzlinu v celkové anestezii |
| | dia_dny | číselník výkonů dnů s chirurgickým výkonem na měkkých nebo pojivových tkáních |
| | hru_dny | číselník výkonů dnů s operačním výkonem v dutině hrudní |
| | bri_dny | číselník výkonů dnů s operačním výkonem v dutině břišní a pánevní |
| | zlu_dny | číselník výkonů dnů s endoskopickým nebo radiologickým výkonem na játrech a žlučových cestách |
| | hrd_dny | číselník výkonů dnů s výkonem hrudní drenáže |
| | oko_dny | číselník výkonů dnů s výkonem na oku |
| | srd_dny | číselník výkonů dnů s výkonem na srdci nebo aortě |
| | cev_dny | číselník výkonů dnů s výkonem na periferních cévách a okolních strukturách |
| | hdl_dny | číselník výkonů dnů eliminačních metod krve |
| | orto_dny | číselník výkonů ortopedie |

| | | |
|---|---|---|
| | polytrauma_dny | číselník operačních výkonů pro polytrauma |
| **Výstupní data** | pgp_04_temp | soubor s položkami hospitalizačních případů, ke kterým je uveden příznak příslušnosti k uvedeným číselníkům |

Na jednotlivé doklady přiřazené k sestaveným hospitalizačním případům ze souboru ***doklady_pripady_temp*** se napojují jednotlivé položky výkonů a DRG markerů, které se pak dohledávají v číselnících (ve složce *<ciselniky>*) kritických výkonů a DRG markerů, terapeutických dní, UPV atd. Ke každé položce se připojuje příznak, zda tato položka náleží do daného číselníku (hmotnost, gestační věk, UPV, terapeutické dny, atd.). V případě DRG markerů pro porodní hmotnost, gestační věk a délku UPV je kontrolováno množství provedení u příslušného výkonu. Pokud odpovídá rozmezí danému tímto markerem, je přesná hodnota převzata z tohoto atributu. Jinak se použije standardní hodnota příslušná tomuto markeru dle METODIKY SESTAVENÍ HOSPITALIZAČNÍCH PŘÍPADŮ.

Kritické ZUP (bez dalšího rozlišení) jsou vyfiltrovány již v rámci procesu PGP_03_POLOZKY.

~~Obrázek 40 Diagram procesu pgp_04b_polozky_ciselniky~~




Výsledek se seřadí a ukládá do dočasného souboru ***pgp_04_temp***.

## pgp_05a_agregace_polozek

*Tabulka 45 Základní popis procesu pgp_05a_agregace_polozek*

| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~HTML – Talend dokumentace~~ | PGP_05A_AGREGACE_POLOZEK | ~~talend_dokumentace\ pgp_05a_agregace_polozek\ pgp_05a_agregace_polozek_2.0.html~~ |
| **Nadřazený job** | PGP_MASTER | |
| **Předchozí job** | PGP_04B_POLOZKY_CISELNIKY | |
| **Následující job** | PGP_05B_AGREGACE_DALSI | |
| **Vstupní data** | pgp_04_temp | soubor sledovaných položek (kritických výkonů, DRG markerů a ZUP) s uvedeným příznakem příslušnosti položky k jednotlivým číselníkům, vzniklý v předchozím procesu PGP_04B_POLOZKY_CISELNIKY |
| **Výstupní data** | pgp_05a_temp | soubor hospitalizačních případů se seznamy kritických výkonů a DRG markerů a kritických ZUP s uvedeným množstvím |

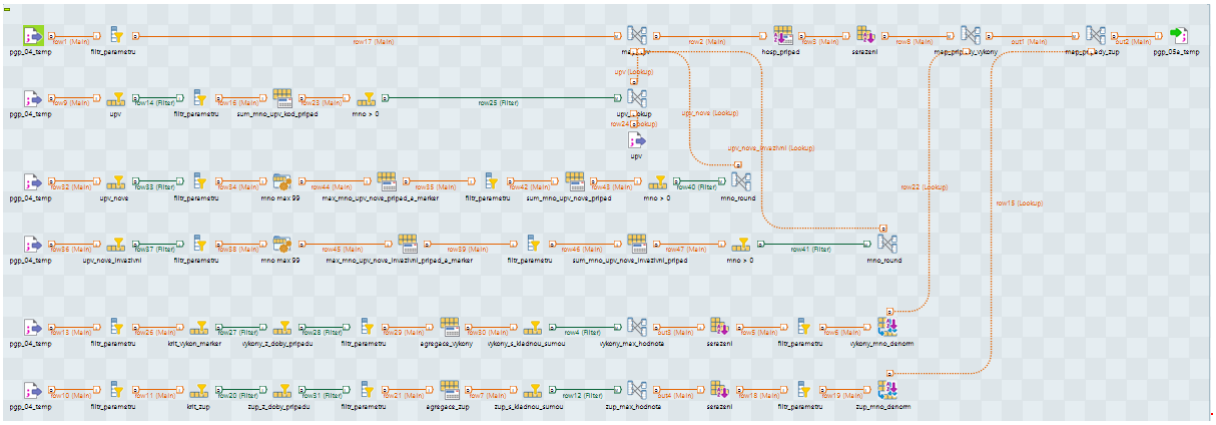
Před samotnou agregací se v první fázi procesu dopočítává hodnota DRG markeru UPV. Hodnota UPV se počítá jen v případě, kdy suma množství na případ je větší než nula. V případě, že množství odpovídá rozmezí uvedenému u daného markeru, přebírá se hodnota přímo z parametru UPV. V opačném případě se použije hodnota pro daný marker přímo z číselníku.

Na agregované záznamy hospitalizačních případů jsou napojeny položky kritických výkonů, DRG markerů a zvlášť účtovaných položek dle příslušných číselníků (ve složce *<ciselniky>*) společně s informací o jejich počtu/množství. Vstupní data jsou načítána ze souboru ***pgp_04_temp*** pro každou z větví samostatně. Jsou vybírány pouze ty kritické výkony a ZUP, které byly provedeny/aplikovány v průběhu hospitalizačního případu. Množství je kontrolováno na maximální počet dle DR a jsou vybírány pouze ty s kladným množstvím. Seznamy kritických výkonů, DRG markerů a kritických ZUP jsou seřazeny dle metodiky (DRG markery před kritickými výkony, dále dle data provedení a kódu položky alfanumericky). Výsledek je ukládán do dočasného souboru ***pgp_05a_temp***.

Ve verzi 5.0 byla provedena úprava kódu pro výpočet parametru *upv* dle změn v metodice sestaveni HP. Byla přidána nová větev pro výpočet celkové délky UPV dle nových markerů na případ, viz *upv_nove*. Dále byla přidána nová větev pro výpočet délky invazivní a neinvazivní UPV (viz v pomocný parametr *upv_nove_invazivni*).

Stará větev pro výpočet referenčního období byla ponechána a kód upraven tak, aby se suma množství pro staré markery počítala ve dnech namísto v hodinách.

~~*Obrázek 41 Diagram procesu pgp_05a_agregace_polozek*~~



## pgp_05b_agregace_dalsi

*Tabulka 46 Základní popis procesu pgp_05b_agregace_dalsi*

| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~**HTML – Talend dokumentace**~~ | PGP_05B_AGREGACE_DALSI | ~~talend_dokumentace\ pgp_05b_agregace_dalsi\ pgp_05b_agregace_dalsi_2.2.html~~ |
| **Nadřazený job** | PGP_MASTER | |
| **Předchozí job** | PGP_05A_AGREGACE_POLOZEK | |
| **Následující job** | PGP_05C_SESTAVENI_VV | |
| **Vstupní data** | pgp_05a_temp | soubor hospitalizačních případů se seznamy kritických výkonů a DRG markerů a kritických ZUP s uvedeným množstvím, vzniklý v procesu PGP_05A_AGREGACE_POLOZEK |
| | pgp_04_temp | soubor sledovaných položek (kritických výkonů, DRG markerů a ZUP) s uvedeným příznakem příslušnosti položky k jednotlivým číselníkům, vzniklý v procesu PGP_04B_POLOZKY_CISELNIKY |
| | validace_pripadu_temp1 | Soubor ID případů a nalezených chyb z procesu PGP_VALIDACE_PRIPADU. Používá se jako filtr pro případy bez chyby typu 3 (chybný nebo nevyplněný věk na prvním dokladu) v komponentě tMap_20 |
| **Výstupní data** | pgp_05b_temp | soubor hospitalizačních případů se seznamy kritických výkonů a DRG markerů, ZUP a dalších sledovaných výkonů včetně jejich množství |
| | validace_pripadu_temp2 | výstupní soubor kontrol hospitalizačních případů na parametry hmotnost, gestacni_vek a upv |
| | terapeuticke_dny | seznam terapeutických dní |
| | pgp_doklady_mimo_HP | od verze 7.0 nový výstupní soubor obsahující doklady s položkou provedenou mimo dobu hospitalizačního případu |

V tomto procesu jsou k hospitalizačním případům ze souboru ***pgp_05a_temp*** doplňovány další údaje ze souboru položek ***pgp_04_temp.*** Agregace počtu propustek na případ je využita při finálním výpočtu parametru LOS, který udává korektní délku hospitalizačního případu dle metodiky. Dále je počítán počet terapeutických dnů pro jednotlivé sledované léčebné modality. Berou se v úvahu jednotlivé typy terapeutických dnů a případné další podmínky pro jejich započtení do terapeutických dnů (celkový čas provedených rehabilitačních výkonů, současné vykázání výkonu celkové anestezie při léčbě popálenin). Výsledek je uložen do souboru ***pgp_05b_temp***.

S určením počtu terapeutických dnů jednotlivých typů je také generován seznam konkrétních kalendářních dnů, kdy byla terapie poskytnuta. Výsledek je ukládán do souboru ***terapeuticke_dny.csv***. Do tohoto souboru se také ukládá nepovinná položka RUN_ID v závislosti na použití parametru runId.

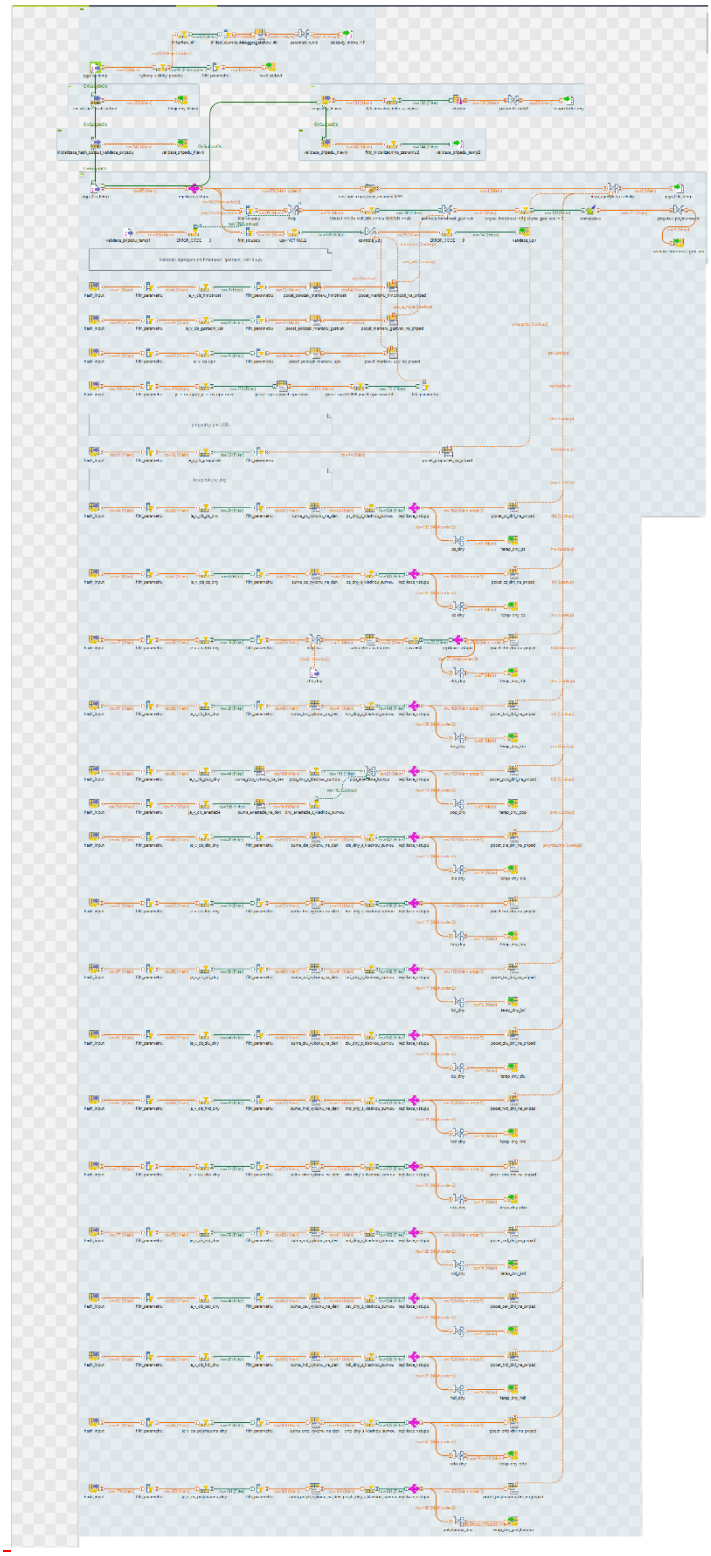
Součástí procesu jsou kontroly na vyplnění údajů o hmotnosti, gestačním věku a UPV vztažené k celému případu – zda je vyplněn alespoň jednou a naopak neobsahuje např. více markerů. Výstup těchto kontrol je ukládán do souboru ***validace_pripadu_temp2.csv***.

Ve verzi 5.0 bylo provedeno napojení pomocného parametru *upv_nove_invazivni* (součet množství dle markerů pro invazivní a neinvazivní UPV) na parametr *KP2*.

Byla přidána větev kontrol pro UPV. Je detekována chyba v případě přítomnosti starých a nových markerů v rámci jednoho případu. Ve vstupní větě se použije stejný kód pro chybu parametru UPV (ERROR_CODE = 9).

Od verze 7.0 byl přidán nový výstupní soubor ***pgp_doklady_mimo_HP***. Tento soubor obsahuje ty položky, které byly provedeny mimo dobu hospitalizačního případu.

~~Obrázek 42 Diagram procesu pgp_05b_agregace_dalsi~~



## pgp_05c_sestaveni_vv

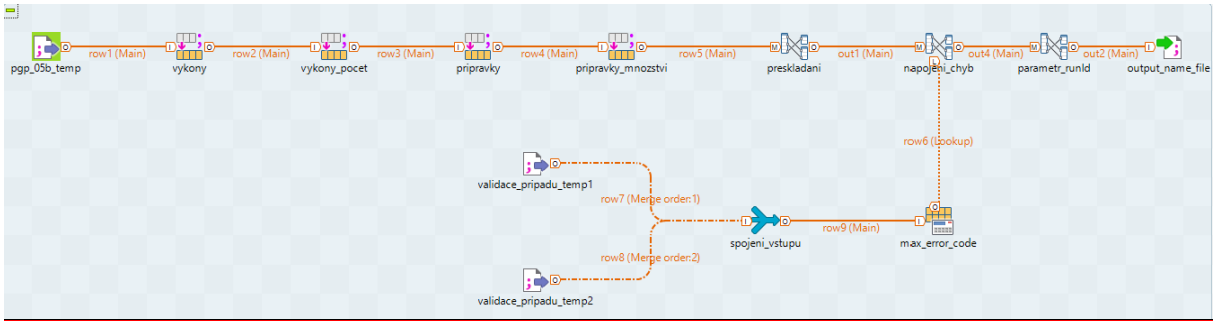
*Tabulka 47 Základní popis procesu pgp_05c_sestaveni_vv*

| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~HTML – Talend dokumentace~~ | PGP_05C_SESTAVENI_VV | ~~talend_dokumentace\pgp_05c_sestaveni_vv\pgp_05c_sestaveni_vv_2.0.html~~ |
| **Nadřazený job** | PGP_MASTER | |
| **Předchozí job** | PGP_05B_AGREGACE_DALSI | |
| **Následující job** | PGP_CLEAR_TEMPDIR | |
| **Vstupní data** | pgp_05b_temp | soubor hospitalizačních případů se seznamy kritických výkonů a DRG markerů, ZUP a dalších sledovaných výkonů včetně jejich množství, vzniklý v procesu PGP_05B_AGREGACE_DALSI |
| | validace_pripadu_temp1 | soubor kontrol na první a poslední doklady hospitalizačních případů z procesu PGP_VALIDACE_PRIPADU |
| | validace_pripadu_temp2 | soubor kontrol na parametry hmotnost, gestacni_vek a upv hospitalizačních případů z procesu PGP_05B_AGREGACE_DALSI |
| **Výstupní data** | <output_name_file> | výsledný soubor vstupních datových vět grouperu dle datového rozhraní |

V procesu jsou seznamy kritických výkonů a zvlášť účtovaných položek společně s informacemi o jejich počtu/množství ze souboru ***pgp_05b_temp*** rozděleny do jednotlivých parametrů datového rozhraní. Následují konečné úpravy a přeskládání vstupní věty do podoby, kterou definuje metodika a datové rozhraní. Hospitalizační případy jsou uloženy do finálního výstupního souboru, jehož pojmenování závisí na nastavení kontextového parametru ***output_name_file*** ve složce definované v parametru *<outdir>*.

K jednotlivým hospitalizačním případům se přidává příznak o nalezené chybě ze souborů ***validace_priapdu_temp1*** a ***validace_pripadu_temp2***. Případy bez chyb mají v parametru chyba_p vyplněnou hodnotu 0. Ostatní mají vyplněnou maximální hodnotu příznaku chyby ze všech nalezených chyb pro daný hospitalizační případ. Také je nastavena nepovinná položka RUN_ID v závislosti na použití parametru runId.

~~*Obrázek 43 Diagram procesu pgp_05c_sestaveni_vv*~~



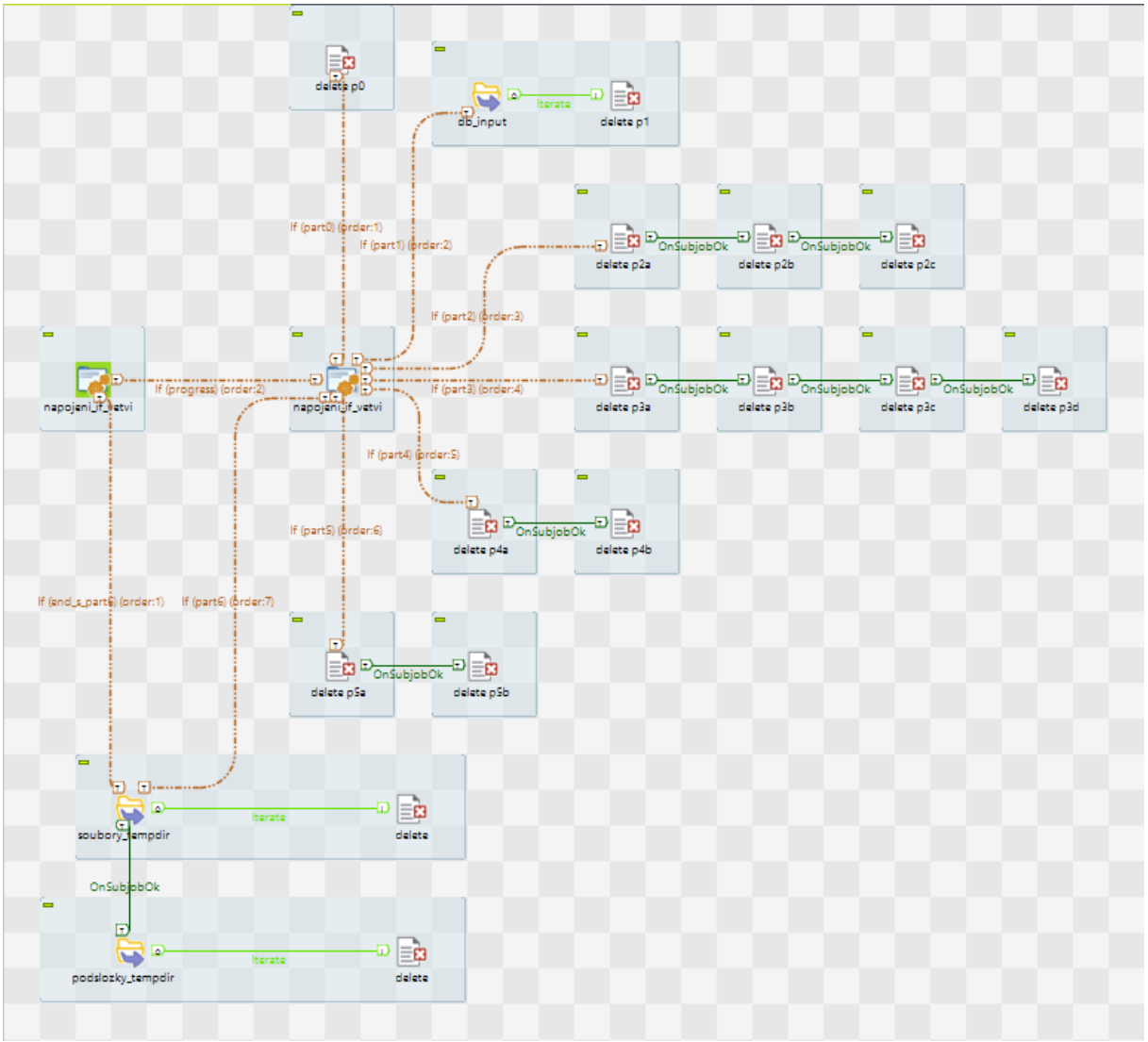
## pgp_clear_tempdir

*Tabulka 48 Základní popis procesu pgp_clear_tempdir*

| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~HTML – Talend dokumentace~~ | PGP_CLEAR_TEMPDIR | ~~talend_dokumentace\pgp_clear_tempdir\pgp_clear_tempdir_2.0.html~~ |
| Nadřazený job | PGP_MASTER | |
| Předchozí job | PGP_05C_SESTAVENI_VV | |
| Následující job | | |
| Vstupní data | - | |
| Výstupní data | - | |

Proces vyčištění složky *<tempdir>* od dočasných souborů a složek vytvořených programem. Nemažou se soubory validačních kontrol, soubor terapeutických dní a soubory fáze napojení dokladů na hospitalizační případy (zde v dokumentu uvedeny v tabulkách ***Validační soubory vytvořené v rámci procesu***).

~~*Obrázek 44 Diagram procesu pgp_clear_tempdir*~~



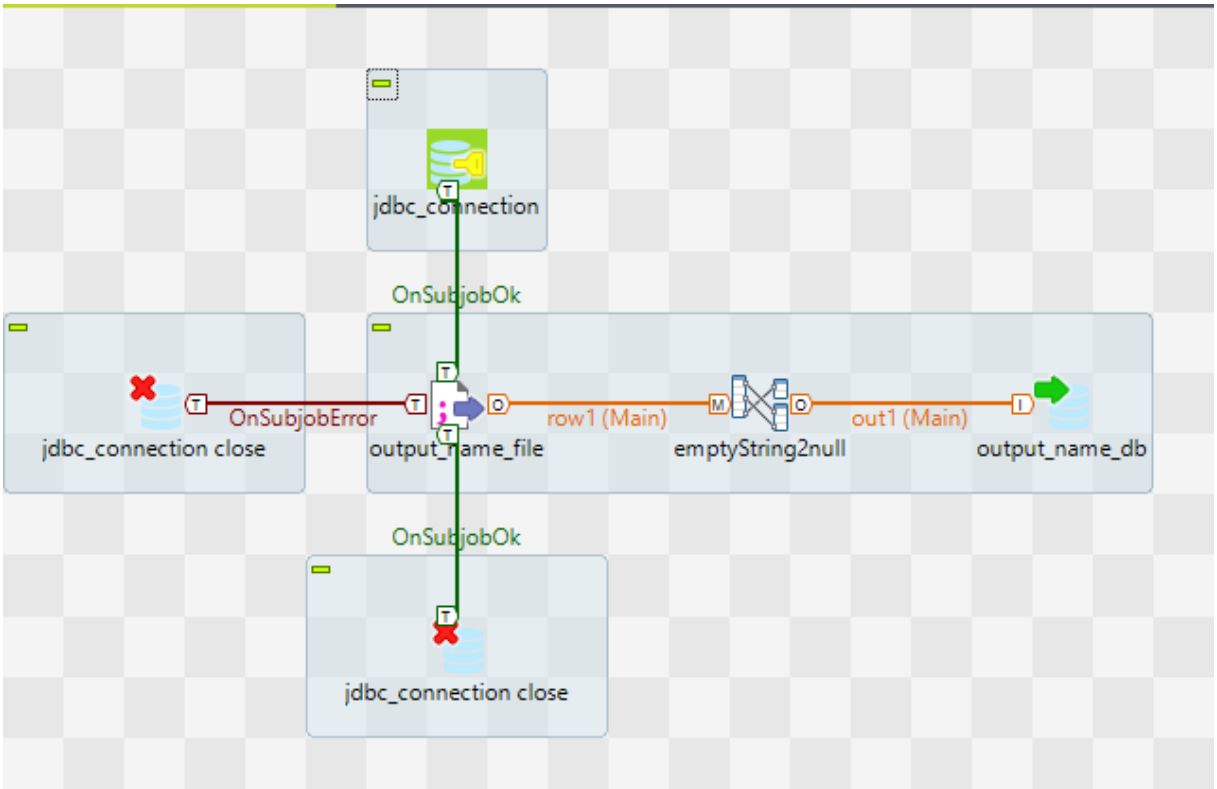
## pgp_db_output

*Tabulka 49 Základní popis procesu pgp_db_output*

| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~HTML – Talend dokumentace~~ | PGP_DB_OUTPUT | ~~talend_dokumentace\pgp_db_output\pgp_db_output_2.0.html~~ |
| **Nadřazený job** | PGP_MASTER | |
| **Předchozí job** | PGP_CLEAR_TEMPDIR | |
| **Následující job** | PGP_SERVICE_KONTROLA_DBDATA | případně poslední proces pre-grouperu (spuštění závisí na hodnotě vstupního parametru db_output) |
| **Vstupní data** | <output_name_file> | výsledný soubor vstupních datových vět grouperu z procesu PGP_05C_SESTAVENI_VV dle datového rozhraní |
| **Výstupní data** | <output_name_db> | databázová tabulka odpovídající datovému rozhraní vstupní datové věty grouperu |

Proces vkládá data z výstupního textového souboru ***<output_name_file>*** do databázové tabulky. Spouští se v závislosti na nastavení hodnoty kontextového parametru ***db_output***. Databázová tabulka musí být předem vytvořena a její název se definuje parametrem ***output_name_db***. U parametrů v textovém formátu se prázdné hodnoty nahrazují za hodnotu *null*. Do databáze se také ukládá nepovinná položka RUN_ID v závislosti na použití parametru runId.

~~*Obrázek 45 Diagram procesu pgp_db_output*~~



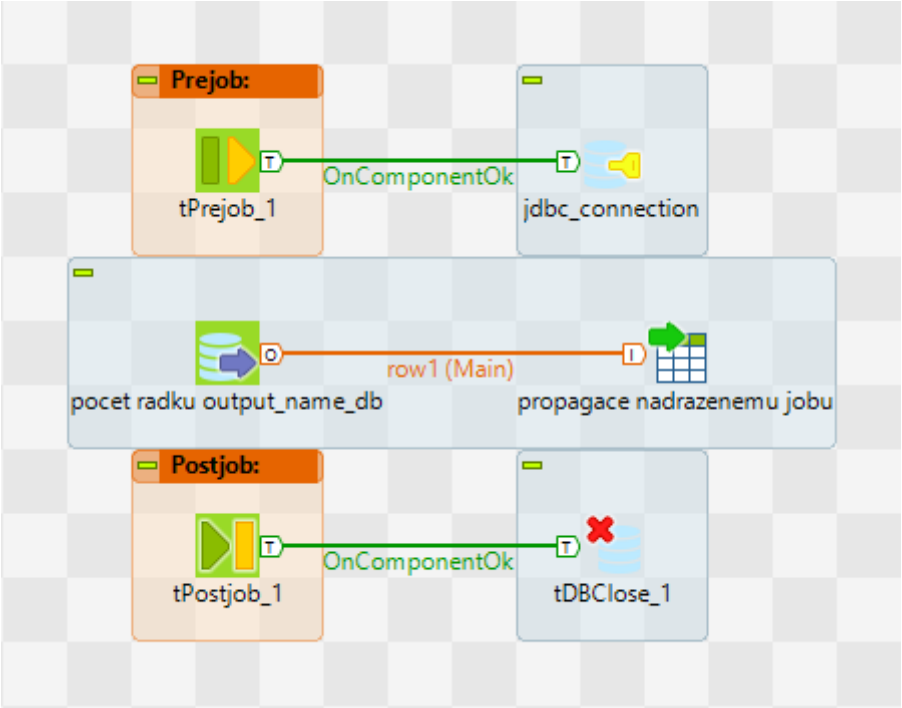
## pgp_service_kontrola_dbdata

*Tabulka 50 Základní popis procesu pgp_service_kontrola_dbdata*

| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~HTML – Talend dokumentace~~ | PGP_SERVICE_KONTROLA_DBDATA | ~~talend_dokumentace\pgp_service_kontrola_dbdata\ pgp_service_kontrola_dbdata_0.4.html~~ |
| **Nadřazený job** | PGP_MASTER | |
| **Předchozí job** | PGP_DB_OUTPUT | |
| **Následující job** | - | poslední proces pre-grouperu (spuštění závisí na hodnotě vstupního parametru service) |
| **Vstupní data** | | |
| **Výstupní data** | | |

Proces se spouští na konci hlavního programu (pgp_master) v případě, že pregrouper běží jako služba. Kontroluje vložení dat do výstupní tabulky v databázi a údaj předává do nadřazeného procesu pgp_service_main. Dle výsledku zpracování služba vrací stav 200 nebo 404.

~~*Obrázek 46 Diagram procesu pgp_service_kontrola_dbdata*~~



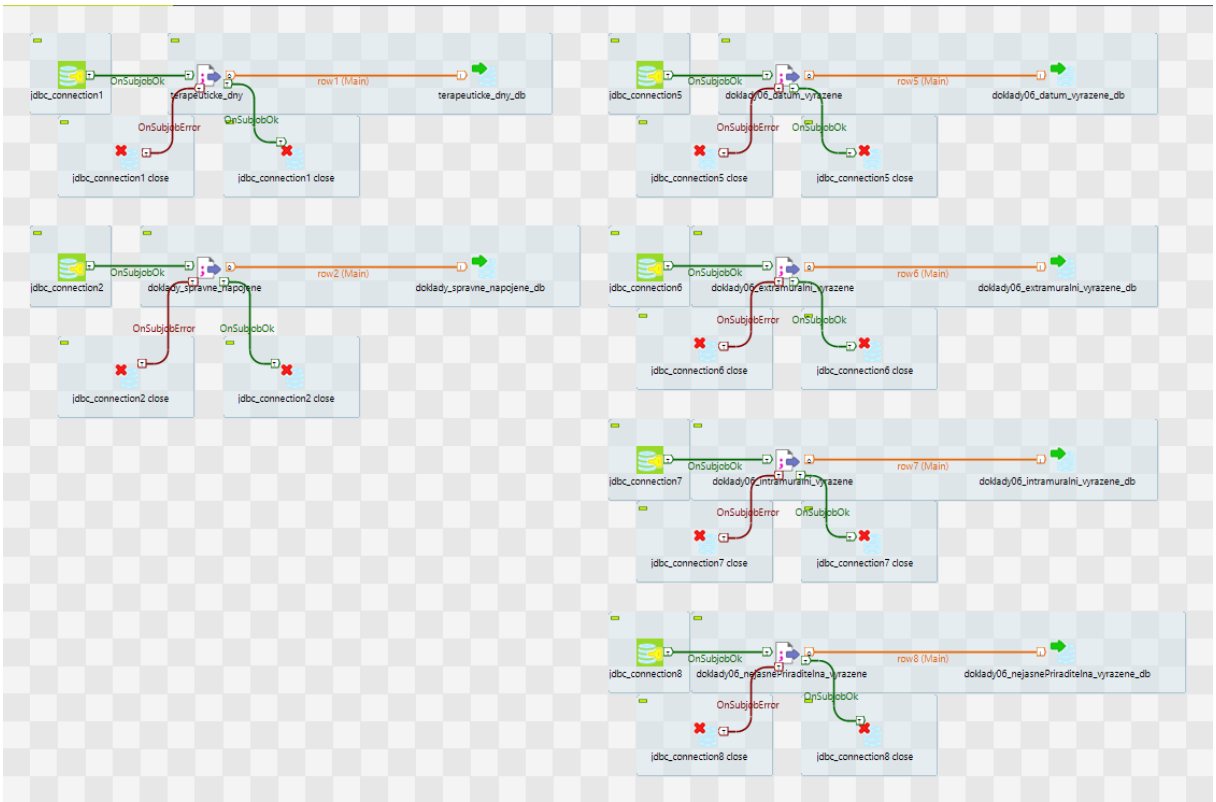
## pgp_temp_db_output

*Tabulka 51 Základní popis procesu pgp_temp_db_output*

| Ukazatel | Hodnota | Poznámka |
|---|---|---|
| ~~**HTML – Talend dokumentace**~~ | PGP_TEMP_DB_OUTPUT | ~~talend_dokumentace\pgp_temp_db_output\pgp_temp_db_output_0.5.html~~ |
| **Nadřazený job** | PGP_MASTER | |
| **Předchozí job** | PGP_CLEAR_TEMPDIR | |
| **Následující job** | - | poslední proces pre-grouperu (spuštění závisí na hodnotě vstupního parametru temp_db) |
| **Vstupní data** | terapeuticke_dny | seznam terapeutických dní z procesu PGP_05B_AGREGACE_DALSI |
| | dokladyPripady_spravneNapojene_filtr | seznem korektně napojených dokladů typu 02 a 06 na hospitalizační případy z procesu SPRAVNENAPOJENE_UNITE |
| | doklady06_datum_vyrazene | seznam nenapojených dokladů na hospitalizační případy podle datumu prvního výkonu z procesu PGP_DOKLADY06PRIPADY_HLAVNI |
| | doklady06_extramuralni_vyrazene | seznam vyřazených dokladů podle pravidel pro extramurální peči z procesu PGP_DOKLADY06PRIPADY_EXTRAMURALNI |
| | doklady06_intramuralni_vyrazene | seznam vyřazených dokladů podle pravidel pro intramurální péči z procesu PGP_DOKLADY06PRIPADY_INTRAMURALNI |
| | doklady06_nejasnePriraditelna_ vyrazene | seznam vyřazených dokladů podle pravidel pro nejasně přiřaditelnou péči z procesu PGP_DOKLADY06PRIPADY_NEJASNEPRIRADITELNA |
| **Výstupní data** | terapeuticke_dny_db | databázová tabulka odpovídající datovému rozhraní |
| | doklady_spravne_napojene_db | databázová tabulka odpovídající datovému rozhraní |
| | doklady06_datum_vyrazene_db | databázová tabulka odpovídající datovému rozhraní |
| | doklady06_extramuralni_vyrazene_db | databázová tabulka odpovídající datovému rozhraní |
| | doklady06_intramuralni_vyrazene_db | databázová tabulka odpovídající datovému rozhraní |
| | doklady06_nejasnePriraditelna_vyrazene_db | databázová tabulka odpovídající datovému rozhraní |

Proces uložení vybraných dočasných souborů procesu do databázových tabulek. Provádí se pouze pokud je nastaven konfigurační parametr ***temp_db*** na hodnotu *true*. Do databáze se také ukládá nepovinná položka RUN_ID v závislosti na použití parametru runId.

~~*Obrázek 47 Diagram procesu pgp_temp_db_output*~~


jdbc_connection1
OnSubjobOk
terapeuticke_dny
row1 (Main)
terapeuticke_dny_db
OnSubjobError
OnSubjobOk
jdbc_connection1 close
jdbc_connection1 close
jdbc_connection2
OnSubjobOk
doklady_spravne_napojene
row2 (Main)
doklady_spravne_napojene_db
OnSubjobError
OnSubjobOk
jdbc_connection2 close
jdbc_connection2 close
jdbc_connection5
OnSubjobOk
doklady06_datum_vyrazene
row5 (Main)
doklady06_datum_vyrazene_db
OnSubjobError
OnSubjobOk
jdbc_connection5 close
jdbc_connection5 close
jdbc_connection6
OnSubjobOk
doklady06_extramuralni_vyrazene
row6 (Main)
doklady06_extramuralni_vyrazene_db
OnSubjobError
OnSubjobOk
jdbc_connection6 close
jdbc_connection6 close
jdbc_connection7
OnSubjobOk
doklady06_intramuralni_vyrazene
row7 (Main)
doklady06_intramuralni_vyrazene_db
OnSubjobError
OnSubjobOk
jdbc_connection7 close
jdbc_connection7 close
jdbc_connection8
OnSubjobOk
doklady06_nejasnePriraditelna_vyrazene
row8 (Main)
doklady06_nejasnePriraditelna_vyrazene_db
OnSubjobError
OnSubjobOk
jdbc_connection8 close
jdbc_connection8 close

# ~~Příloha 1: Testování výstupů pre-grouperu~~

## ~~Přiřazení dokladů 02 k hospitalizačním případům~~

| ~~Číslo testu~~ | ~~Název testu~~ |
|---|---|
| ~~1~~ | ~~Neexistuje doklad 02 přiřazený k HP, který by začínal dříve než HP~~ |
| ~~2~~ | ~~Existuje doklad 02 přiřazený k HP, který začíná ve stejný den jako HP~~ |
| ~~3~~ | ~~Neexistuje doklad 02 přiřazený k HP, který by končil později než HP~~ |
| ~~4~~ | ~~Existuje doklad 02 přiřazený k HP, který končí ve stejný den jako HP~~ |
| ~~5~~ | ~~Všechny doklady 02 s odborností akutního lůžka a vyplněnými povinnými údaji jsou přiřazeny k nějakému HP~~ |
| ~~6~~ | ~~Nevyskytují se doklady 02 přiřazené k HP, které mají odbornost mimo akutní lůžka~~ |
| ~~7~~ | ~~Nevyskytují se současně doklady 02 odbornosti 2H1/2F1 a jakékoliv jiné v rámci HP~~ |
| ~~8~~ | ~~Hlavní diagnóza HP odpovídá základní DG nějakého dokladu 02 končícího ve stejný den~~ |
| ~~9~~ | ~~Vedlejší diagnózy HP odpovídají vedlejším DG nějakého dokladu 02 končícího ve stejný den~~ |
| ~~10~~ | ~~Přijímací odbornost HP odpovídá odbornosti nějakého dokladu 02 začínajícího ve stejný den jako HP~~ |
| ~~11~~ | ~~Propouštěcí odbornost HP odpovídá odbornosti nějakého dokladu 02 končícího ve stejný den jako HP~~ |
| ~~12~~ | ~~Doporučení k hospitalizaci u HP odpovídá doporučení k hospitalizaci nějakého dokladu 02 začínajícího ve stejný den jako HP~~ |
| ~~13~~ | ~~Není doklad 02 u HP, který začíná ve stejný den jako HP a má doporučení k hospitalizaci s vyšší prioritou~~ |
| ~~14~~ | ~~Způsob ukončení hospitalizace u HP odpovídá způsobu ukončení nějakého dokladu 02 končícího ve stejný den jako HP~~ |
| ~~15~~ | ~~Není doklad 02 u HP, který končí ve stejný den jako HP a má způsob ukončení hospitalizace s vyšší prioritou~~ |
| ~~16~~ | ~~K HP není přiřazen doklad 02 jiného ZZ než ZZ HP~~ |

## ~~Přiřazení dokladů 06 k hospitalizačním případům~~

| ~~Číslo testu~~ | ~~Název testu~~ |
|---|---|
| ~~17~~ | ~~K HP jsou přiřazeny pouze doklady 02 a 06~~ |
| ~~18~~ | ~~Alespoň jedno ZZ na dokladu 06 (provádějící nebo žádající) musí odpovídat ZZ HP~~ |
| ~~19~~ | ~~První výkon na dokladu 06 spadá do období HP~~ |
| ~~20~~ | ~~HP akutní rehabilitace v jednom ZZ se nesmějí zcela překrývat~~ |
| ~~21~~ | ~~HP akutní rehabilitace v jednom ZZ se nesmí částečně překrývat nebo být přerušeny na nejvýše 1 celý kalendářní den~~ |
| ~~22~~ | ~~HP mimo akutní rehabilitace v jednom ZZ se nesmějí zcela překrývat~~ |
| ~~23~~ | ~~HP mimo akutní rehabilitace v jednom ZZ se nesmí částečně překrývat nebo být přerušeny na nejvýše 1 celý kalendářní den~~ |
| ~~24~~ | ~~Doklad 06 není přiřazen jako extramurál (přes ZZ žádajícího) k jednomu HP, pokud může být přiřazen jako intramurál (resp. nejasně přiřaditelná péče přes ZZ provádějícího) k jinému HP~~ |
| ~~25~~ | ~~Doklad 06 nezůstane nepřiřazen k žádnému HP, pokud může být přiřazen jako intramurál k nějakému HP~~ |
| ~~26~~ | ~~Doklad 06 nezůstane nepřiřazen k žádnému HP, pokud může být přiřazen jako nejasně přiřaditelná péče k nějakému HP~~ |
| ~~27~~ | ~~Doklad 06 nezůstane nepřiřazen k žádnému HP, pokud může být přiřazen jako extramurál k nějakému HP~~ |

| | |
|---|---|
| ~~28~~ | ~~Žádný doklad 06 přiřazený k HP neobsahuje odbornosti vyloučené z přiřazení k HP~~ |

## ~~Sestavení hospitalizačního případu~~

| ~~Číslo testu~~ | ~~Název testu~~ |
|---|---|
| ~~29~~ | ~~HP akutní rehabilitace u jednoho ZZ se celé nepřekrývají~~ |
| ~~30~~ | ~~HP akutní rehabilitace u jednoho ZZ se nepřekrývají z části ani nejsou přerušeny na max. 1 kalendářní den~~ |
| ~~31~~ | ~~HP mimo akutní rehabilitace u jednoho ZZ se celé nepřekrývají~~ |
| ~~32~~ | ~~HP mimo akutní rehabilitace u jednoho ZZ se nepřekrývají z části ani nejsou přerušeny na max. 1 kalendářní den~~ |

## ~~Náplň a vlastnosti hospitalizačního případu~~

| ~~Číslo testu~~ | ~~Název testu~~ |
|---|---|
| ~~33~~ | ~~HP obsahuje všechny kritické výkony a DRG markery, které jsou na přiřazených dokladech~~ |
| ~~34~~ | ~~HP neobsahuje jiné než kritické výkony a DRG markery, které jsou na přiřazených dokladech~~ |
| ~~35~~ | ~~HP má správně vypočteno množství kritických výkonů a DRG markerů~~ |
| ~~36~~ | ~~Každý kritický výkon a DRG marker se vyskytuje na HP nejvýše jednou~~ |
| ~~37~~ | ~~HP obsahuje všechny kritické ZUP, které jsou na přiřazených dokladech~~ |
| ~~38~~ | ~~HP neobsahuje jiné než kritické ZUP, které jsou na přiřazených dokladech~~ |
| ~~39~~ | ~~HP má správně vypočteno množství kritických ZUP~~ |
| ~~40~~ | ~~Každý kritický ZUP se vyskytuje na HP nejvýše jednou~~ |
| ~~41~~ | ~~Pro každý parametr počtu terapeutických dní (ozařovacích dní, rehabilitace, akutní psychiatrické péče, …) je uvedena správná hodnota~~ |
| ~~42~~ | ~~Počet dnů na propustce a doba los nesmí přesáhnout počet kalendářních dní od začátku do konce případu + 1~~ |
| ~~43~~ | ~~Počet dnů bez hospitalizačního dokladu a doba los nesmí přesáhnout počet kalendářních dní od začátku do konce případu + 1~~ |
| ~~44~~ | ~~V seznamu kritických výkonů a DRG markerů HP se nevyskytuje marker pro UPV, porodní hmotnosti a gestačního věku novorozence~~ |
| ~~45~~ | ~~Pokud byl na dokladu 02 přiřazeném k HP vykázán marker UPV, je vyplněn parametr délky UPV u HP~~ |
| ~~46~~ | ~~Pokud byl na dokladu 02 přiřazeném k HP vykázán marker porodní hmotnosti novorozence, je vyplněn parametr porodní hmotnosti u HP~~ |
| ~~47~~ | ~~Pokud byl na dokladu 02 přiřazeném k HP vykázán marker gestačního věku novorozence, je vyplněn parametr gestačního věku u HP~~ |
| ~~48~~ | ~~Žádné dva HP nemají stejný identifikátor~~ |