

LightComp ScanFinalizer

Verze 1.0.0.17

Dokumentace

Obsah

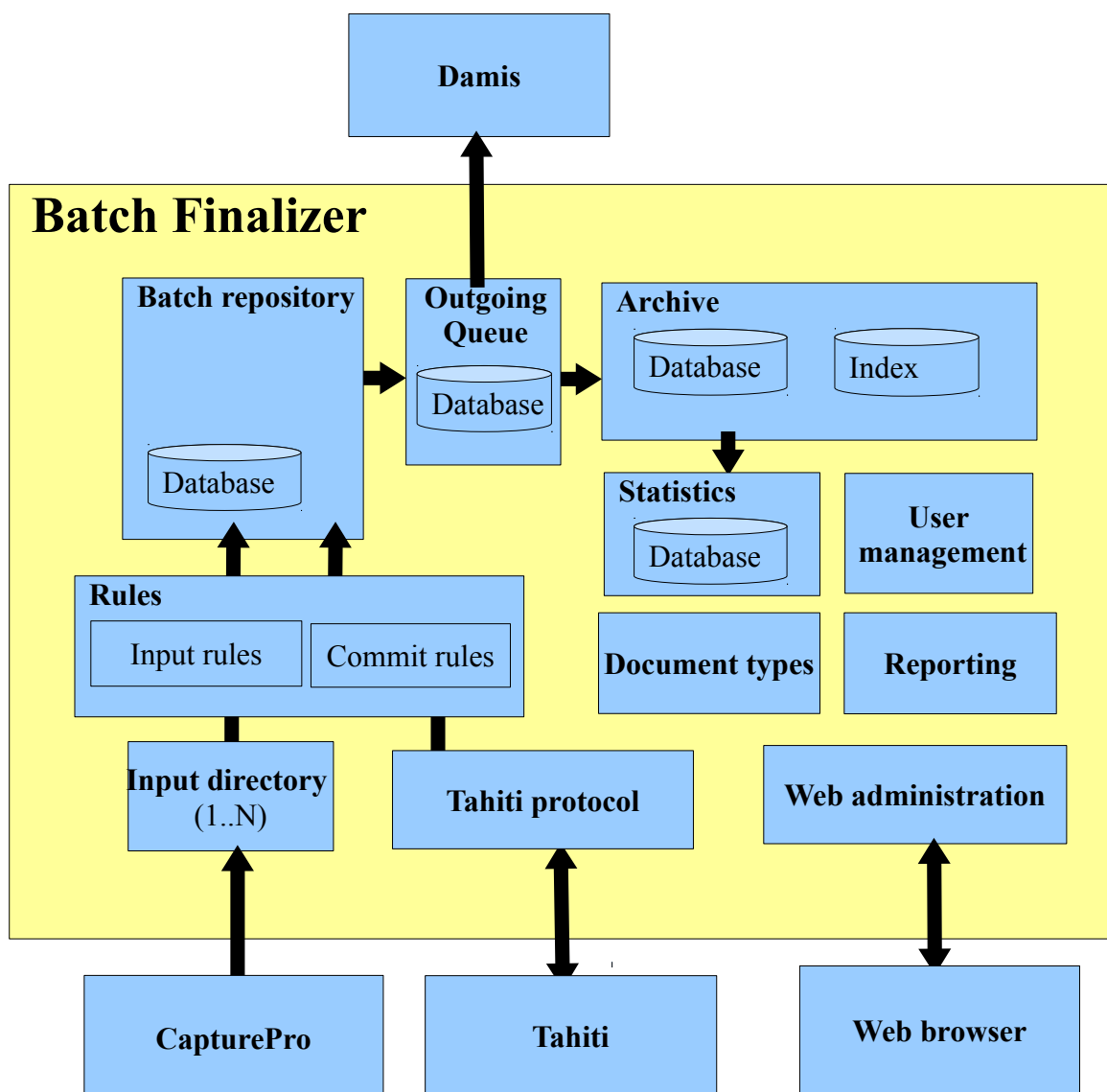
1 Úvod.....	3
2 Architektura systému.....	4
3 Komponenty.....	5
3.1 Batch repository.....	5
3.2 Archive	5
3.3 Outgoing Queue.....	6
3.4 Rules.....	6
3.5 Input directory.....	6
3.6 User management.....	6
3.6.1 Oprávnění.....	6
3.7 Statistics.....	7
3.8 TahitiProtocol.....	7
3.9 Reporting.....	7
3.10 Document types.....	7
4 Průchod dávky finalizační komponentou.....	8
4.1 Kompletní úspěšný průchod dávky systémem.....	8
4.2 Odmítnutí dávky Damisem.....	8
4.3 Vymazání dávky.....	9
4.4 Odemknutí dávky.....	9
4.5 Odmítnutí dávky CommitRules.....	9
5 Rules.....	10
5.1 Vstupní objekty	10
5.2 Výstupní objekty.....	10
5.2.1 setFail.....	10
5.2.2 setDocumentFail.....	10
5.3 Pomocné objekty.....	10
5.4 Příklad.....	11
5.4.1 Vysvětlení.....	12
5.5 Testování pravidel.....	12
5.5.1 Příprava testovacích dat.....	12
5.5.1.1 Příklad testovacích dat.....	13
5.5.1.2 Testování pomocí testovacích dat.....	14
6 Konfigurace.....	16
6.1 Tahiti 6.....	16

1 Úvod

V tomto dokumentu je popsána vnitřní architektura systému LightComp ScanFinalizer a jeho chování při zpracování dávky a způsob konfigurace.

2 Architektura systému

Na obrázku 1 je zobrazena vnitřní architektura systému ScanFinalizer. Celý systém je pak zobrazen v kontextu všech systémů spolupracujících v procesu skenování a finalizace. Šípkami jsou vyznačeny toky dat v průběhu zpracování dávky.



3 Komponenty

Scan Finalizer se skládá z následujících komponent:

- Batch repository
- Archive
- Input directory
- Rules
- User management
- Statistics
- Tahiti protocol
- Document types

Pro svou činnost potřebuje ScanFinalizer následující externí komponenty:

- Tahiti
- Damis
- CapturePro
- Web browser

3.1 Batch repository

Komponenta BatchRepository obsahuje dávky přečtené ze vstupního adresáře, které čekají na potvrzení od operátora. Dávky jsou uloženy v interním repositáři na disku a záznam o stavu každé dávky je v databázi. Dávka se v průběhu zpracování může nacházet v následujících stavech:

1. **Invalid** – dávka čeká na zpracování, po vložení byla označena jako **Invalid**, protože nesplňuje některé z požadovaných kritérií, v tomto stavu může být také dávka, která byla odmítnuta navazujícím systémem Damis
2. **Validate** – dávka čeká na zpracování, vstupní kontrola dávky nezjistila žádné problémy
3. **Locked** – dávku má zamknutou konkrétní uživatel a právě ji zpracovává v Tahiti
4. **Sending** – dávka byla zkontrolována a právě je odesílána do systému Damis
5. **Sent** – dávka byla úspěšně odeslána do systému Damis a právě probíhá mazání dat dávky
6. **Delete** – dávka byla určena k vymazání, vymazání proběhne na pozadí

3.2 Archive

V komponentě **Archive** jsou zaarchivována všechna použitá čísla dávek, čísla jednacích a soubor s popisem každého spisu (docset.xml). Veškeré atributy jsou indexovány, takže je možné podle jejich hodnot vyhledávat. Indexování atributů probíhá vždy až po úspěšném odeslání dávky do Damisu.

3.3 Outgoing Queue

Komponenta **OutgoingQueue** se stará o odeslání potvrzené dávky do Damisu. Pokud je Damis nedostupný pokouší se dávku odeslat dokud se to nepovede. Záznam o odesílané dávce je v databázi a dávka je v průběhu odesílání ve stavu **Sending**.

3.4 Rules

Komponenta **Rules** poskytuje systém pravidel pro kontrolu správnosti spisu. Správnost spisu se kontroluje při vstupu spisu do systému a před označením spisu za zkontrolovaný.

- **InputRules** – kontrolují dávku při vstupu do systému, pokud dávka nevyhoví nějakému kritériu, tak je vložena ve stavu **Invalid**.
- **CommitRules** – kontrolují dávku při potvrzení uživatelem, pokud dávka nesplňuje některá kritéria, tak se ji nepodaří potvrdit a uživateli se zobrazí chybová hláška s důvodem odmítnutí.

3.5 Input directory

Vstupní adresář, do kterého jsou vkládány naskenované dávky určené k finalizaci. Adresář je v pravidelných intervalech monitorován na přítomnost nové dávky. Pokud je nová dávka nalezena, tak je zkontrolována její struktura. Kontrolují se následující vlastnosti:

- Existence všech souborů obsažených v popisu spisu
- Nenulová délka všech souborů
- Existence jednoznačného identifikátoru u všech dokumentů spisu
- Existence jednoznačného identifikátoru dávky u všech spisů
- Existence povinných atributů spisu

3.6 User management

Komponenta pro správu uživatelů. Uživatelé zde mohou být vytvářeni, mazáni a mohou jim být měněna oprávnění.

3.6.1 Oprávnění

V systému jsou definována následující oprávnění:

- **Admin** – uživatel s tímto oprávněním může vše
- **Commit** – uživatel s tímto oprávněním může odesílat zkontrolované dávky, v Tahiti vidí pouze své dávky
- **CommitAll** – uživatel s tímto oprávněním může odesílat zkontrolované dávky přiřazené libovolnému uživateli, v Tahiti vidí dávky všech uživatelů
- **Delete** – uživatel s tímto oprávněním může mazat dávky, v Tahiti vidí pouze své dávky
- **DeleteAll** – uživatel s tímto oprávněním může mazat dávky přiřazené libovolnému uživateli, v Tahiti vidí dávky všech uživatelů
- **Search** – uživatel s tímto oprávněním může vyhledávat dávky

- **Statistics** – uživatel s tímto oprávněním může prohlížet statistiky

3.7 Statistics

Statistický modul. Zde se uchovávají statistiky počtu zpracovaných stránek jednotlivými uživateli. Počty stránek jsou evidovány po následujících kategoriích:

- Prepared – počet stránek zpracovaných přípravičem
- Operated – počet naskenovaných stránek
- Finalized – počet zkontrolovaných stránek

3.8 TahitiProtocol

Komponenta zajišťující komunikaci mezi Finalizací a Tahiti. Poskytuje Tahiti následující služby:

1. Autorizaci uživatelů
2. Konfigurační soubory pro Tahiti
3. Operace související se zpracováním dávky

3.9 Reporting

Komponenta **Reporting** zajišťuje generování tiskových sestav s popisem zpracované dávky. Report se generuje automaticky po úspěšné kontrole dávky před jejím odesláním. Pokud je dávka odesílána z Tahiti je vygenerovaný report poslán do Tahiti (je tam poslána adresa, ze které si může report stáhnout) a report je zobrazen ve webovém prohlížeči. Pokud je dávka odesílána z webového rozhraní, je report ihned zobrazen v novém okně prohlížeče. Pro vytvoření reportu jsou k dispozici následující proměnné:

Název proměnné	Význam
BatchId	Id dávky
Prepare	Jméno přípraviče dávky
Operate	Jméno skenera dávky
Finalize	Jméno uživatele, který finalizoval dávku
totalPages	Celkový počet stránek dávky

3.10 Document types

Komponenta **Document types** zajišťuje správu dokumentových typů používaných v průběhu zpracování. Dokumentové typy mohou být používány statické nahrávané z disku nebo mohou být průběžně stahovány z externího systému přes http (např. z Damisu).

4 Průchod dávky finalizační komponentou

4.1 Kompletní úspěšný průchod dávky systémem

Dále je popsáno zpracování jedné dávky od vložení do vstupního adresáře až po odeslání do Damisu v jednotlivých po sobě jdoucích krocích.

1. **Input directory** zjistí novou dávku ve vstupním adresáři.
2. Provede se kontrola formátu vstupní dávky. Pokud je dávka vadná bude vložena do systému ve stavu **Invalid**
3. Dávka je zkontrolována na duplicitu identifikátorů proti identifikátorům uložených v komponentě Archive.
4. Provede se kontrola pomocí **Input rules**. Pokud některý spis nespĺňuje požadovaná kritéria bude dávka vložena do systému ve stavu **Invalid**
5. Data dávky jsou zkopírována do **BatchRepository**, zde je vytvořen záznam v databázi o dávce. Stav dávky je **Validate** nebo **Invalid** podle výsledku předchozích kontrol.
6. Data dávky jsou vymazána ze vstupního adresáře
7. Uživatel si vybere dávku v Tahiti. Dávka je přepnuta do stavu **Locked**.
8. Uživatel dávku zkontroluje a provede commit.
9. Dávka je zkontrolována pomocí **Commit rules**. Pokud dávka nespĺní požadovaná kritéria, může uživatel dávku buď dopravit, nebo vymazat, nebo přiřadit jinému uživateli.
10. K potvrzené dávce je vytvořen záznam o použitých identifikátorech v komponentě Archive (tím se zabrání opětovnému potvrzení dávky se stejnými identifikátory). Dále jsou do databáze uloženy záznamy o jednotlivých spisech (docset.xml).
11. Do Tahiti je poslána zpráva, aby zobrazilo ve webovém prohlížeči pdf soubor s popisem dávky.
12. Stav dávky je změněn na **Sending** a dávka je vložena do odesílací fronty.
13. Po úspěšném odeslání do Damisu je stav dávky změněn na **Sent**.
14. Do komponenty **Archive** jsou uloženy záznamy o jednotlivých spisech (docset.xml).
15. Archivovaná dávka je zaindexována.
16. Data dávky jsou vymazána z interního repositáře.
17. Podle obsahu zaarchivované dávky jsou upraveny statistiky.

4.2 Odmítnutí dávky Damisem

Dále je popsána situace, kdy je dávka odmítnuta Damisem z důvodu kolize identifikátorů. Popis navazuje na předchozí scénář v bodě 13

13. Dávka je odmítnuta Damisem, její stav je změněn na **Invalid**
14. Z komponenty **Archive** jsou vymazány záznamy o použitých identifikátorech dávky a spisů
15. Do chybového stavu dávky je uložena informace, proč byla odmítnuta

4.3 Vymazání dávky

Dále je popsána situace, kdy je dávka vymazána uživatelem. Popis navazuje na předchozí scénář v bodě 10

10. Stav dávky je změněn na **Delete**
11. Data dávky jsou na pozadí vymazána.
12. Záznam o dávce je odstraněn z databáze

4.4 Odemknutí dávky

Dále je popsána situace kdy uživatel odemkne dávku. Popis navazuje na popis úspěšného průchodu po bodu 7

8. Uživatel odemkne dávku v Tahiti
9. Dávka je zkontrolována pomocí **CommitRules** a výsledek je uložen do databáze. Pokud si dávku znova otevře popis chyb odpovídá současnému stavu rozpracovanosti dávky.
10. Dávka je přepnuta do stavu **Invalid** nebo **Validate** podle výsledku kontroly v předchozím bodě.

4.5 Odmítnutí dávky **CommitRules**

Dále je popsána situace, kdy finalizace nepovolí odeslání zkontrolované dávky kvůli nesplnění kontrolních pravidel. Popis navazuje na popis úspěšného průchodu v bodu 9.

9. Dávka je odmítnuta kvůli nesplnění pravidel v **CommitRules**.
10. Do databáze se zapíše aktualizovaný stav kontroly dávky

5 Rules

Pro vyhodnocení požadovaných vlastností jednotlivých spisů je použit systém Drools (<http://www.jboss.org/drools>). Pravidla jsou vyhodnocována na každém jednotlivém spisu vstupujícím do systému nebo zkontrolovaném uživatelem a připraveném k odeslání.

Pro rozhodování do systému vstupují objekty na základě jejichž vlastností rozhodování probíhá a dále objekty do nichž se zapisuje výsledek rozhodnutí. Vstupním objektů v terminologii Drools odpovídají *asserts*, výstupním a pomocným objektům *globals*.

5.1 Vstupní objekty

Vstupním objektem je objekt *DamisDocument* reprezentující jeden dokument ze spisu. Pro rozhodování je k dispozici seznam všech atributů dokumentu.

```
public class DamisDocument {
    Map<String,String> attributes;
};
```

5.2 Výstupní objekty

Výstupním objektem určeným pro uložení výsledku rozhodování je objekt *DocSetStatus*.

```
public class DocSetStatus {
    public void setFail(String problemDescription) ;
    public void setDocumentFail(DamisDocument doc, String problemDescription);
};
```

Dále je popsán význam jednotlivých metod

5.2.1 setFail

Metoda *setFail* způsobí, že celý spis je označen za chybný. Textový řetězec předávaný jako parametr je přidán do seznam chyb spisu. V Tahiti budou všechny spisy dokumentu považovány za chybné a budou tak vyznačené v uživatelském rozhraní.

5.2.2 setDocumentFail

Metoda *setDocumentFail* způsobí, že dokument je označen za chybný. Textový řetězec předaný jako parametr bude přidán do seznamu chyb dokumentu. V Tahiti je takovýto dokument vyznačen v uživatelském rozhraní a seznam chyb je dostupný v podobě tooltipu po najetí kurzorem myši na ikony příslušného dokumentu.

5.3 Pomocné objekty

Pomocným objektem je objekt *CMServer2* obsahující mapování zkratky dokumentového typu na reálný dokumentový typ. Vlastnost *shortNames* obsahuje mapování zkratky dokumentového typu na reálný dokumentový typ.

```
public class CMServer2 {
    Map<String,String> shortNames;
```

```
};
```

5.4 Příklad

```
package finalizerinputrules;
import com.lightcomp.finalizer.damisdocumentset.DamisDocument;
global com.lightcomp.finalizer.batchvalidation.DocSetStatus state;
global com.lightcomp.finalizer.documenttypes.cmserver2.CMServer2 docTypes;

rule "Document type"
when
    d: DamisDocument()
then
    String docCategory = d.getAttribute("Document.Category");
    String docType = (String)docTypes.getShortNames().get(docCategory);
    if ( docType != null ) {
        d.setAttribute("Document.type",docType);
    } else {
        if ( docCategory == null ) {
            state.setFail("Missing Document.category");
        } else {
            state.setFail("Unknown document category:"+docCategory);
        }
    }
}
end

rule "Nerozpoznany druh ps"
when
    d: DamisDocument(attributes["POZNAMKA"]=="DruhPS#000000#DruhPS")
then
    state.setFail("Nerozpoznany druh ps");
end

rule "Kontrola RCICOSPZ"
when
```

```
d: DamisDocument( attributes["RCICOSPZ"]=="0000000000" ||
    ( availableAttributes contains "RCICOSPZ" &&
      attributes["RCICOSPZ"] not matches ".{10}" ) )
then
    state.setDocumentFail(d,"Invalid RCICOSPZ");
end
```

5.4.1 Vysvětlení

Pravidla mají dvě sekce *when* a *then*. Pokud je splněna podmínka v sekci *when* vykoná se sekce *then*

- „**Document type**“ - pravidlo je aktivováno pro každý dokument spisu, v sekci *then* se nastaví dokumentový typ, pokud dokument neobsahuje atribut `Document.Category` je spis označen jako chybný s popisem chyby "Missing Document.category", pokud k hodnotě atributu `Document.Category` neexistuje dokumentový typ je spis označen jako chybný s popisem chyby "Unknown document category:"
- "**Nerozpoznany druh ps**" – pravidlo je aktivováno pokud dokument obsahuje atribut `POZNAMKA` s hodnotou "DruhPS#000000#DruhPS", v sekci *then* se nastaví celý spis jako chybný s popisem chyby "Nerozpoznany druh ps".
- "**Kontrola RCICOSPZ**" – pravidlo je aktivováno pokud v dokumentu existuje atribut `RCICOSPZ` s hodnotou "0000000000" nebo existuje atribut `RCICOSPZ` s hodnotou která není složena z deseti znaků, dokument je pak v sekci *then* označen jako chybný s popisem chyby "Invalid RCICOSPZ".

5.5 Testování pravidel

Pravidla mohou být před použitím testována. Vývoj a testování pravidel probíhá následujícím způsobem:

1. Příprava testovacích dat
2. Vývoj nových pravidel nebo úprava stávajících
3. Test pomocí testovacích dat
4. Pokud jsou v testu zjištěny chyby zpět na bod 2)

5.5.1 Příprava testovacích dat

Testovací data jsou ve formátu `docset.xml`. Každý dokument obsahuje jeden testovaný jev. Kromě standardních atributů dokumentu jsou do každého dokumentu přidány atributy využitě při testování.

Atribut	Význam
Finalize.test.description	Popis testu. Tento popis je zobrazen v uživatelském rozhraní aplikace.
Finalize.test.expectedResult	Očekávaný výsledek testu. 0-bez chyby,1-chyba.

Table 1: Atributy testování

5.5.1.1 Příklad testovacích dat

Příklad testovacích dat slouží k testování správnosti dokumentů z oblasti POJ_UDAL a MAJ_UDAL. U každého dokumentu je vyžadován atribut CPS_CPU obsahující 10 číslic. Pokud atribut obsahuje „0000000000“ je to příznak chybně vytěženého CPS_CPU.

```
<?xml version="1.0"?>
<DocumentSet>
  <Document>
    <Attributes>
      <Attribute name="Document.Category" value="MAJ_UDAL-007"/>
      <Attribute name="CPS_CPU" value="9876543210"/>
      <Attribute name="Finalize.test.expectedResult" value="0"/>
      <Attribute name="Finalize.test.description" value="Valid - MAJ_UDAL,POJ_UDAL"/>
    </Attributes>
    <Files>
      <File name="001_00001.tif" mimeType="image/tiff"/>
    </Files>
  </Document>
  <Document>
    <Attributes>
      <Attribute name="Document.Category" value="MAJ_UDAL-007"/>
      <Attribute name="Finalize.test.expectedResult" value="1"/>
      <Attribute name="Finalize.test.description" value="Missing CPS_CPU"/>
    </Attributes>
    <Files>
      <File name="001_00001.tif" mimeType="image/tiff"/>
    </Files>
  </Document>
  <Document>
```

```
<Attributes>
  <Attribute name="Document.Category" value="MAJ_UDAL-007"/>
  <Attribute name="CPS_CPU" value="987654321"/>
  <Attribute name="Finalize.test.expectedResult" value="1"/>
  <Attribute name="Finalize.test.description" value="Invalid CPS_CPU only 9 digits"/>
</Attributes>
<Files>
  <File name="001_00001.tif" mimeType="image/tiff"/>
</Files>
</Document>
<Document>
  <Attributes>
    <Attribute name="Document.Category" value="MAJ_UDAL-007"/>
    <Attribute name="CPS_CPU" value="0000000000"/>
    <Attribute name="Finalize.test.expectedResult" value="1"/>
    <Attribute name="Finalize.test.description" value="Invalid CPS_CPU ten zeroes"/>
  </Attributes>
  <Files>
    <File name="001_00001.tif" mimeType="image/tiff"/>
  </Files>
</Document>
</DocumentSet>
```

nazdar

Příklad obsahuje 4 testovací případy:

1. Správný dokument, je očekáváno, že nebude pravidly odmítnut.
2. Dokument s chybějícím CPS_CPU, je očekáváno, že bude pravidly odmítnut.
3. Dokument s CPS_CPU obsahujícím pouze 9 číslic, je očekáváno, že bude pravidly odmítnut.
4. Dokument s CPS_CPU obsahujícím „000000000“ , je očekáváno, že dokument bude pravidly odmítnut.

5.5.1.2 Testování pomocí testovacích dat

Testovací data je možné vyhodnotit pomocí konkrétního pravidla z webového rozhraní aplikace. Výsledkem je seznam jednotlivých testovacích případů a jestli se výsledek vyhodnocení každého dokumentu shoduje s očekávaným výsledkem. Při vyhodnocování se aplikují pouze vstupní pravidla, nedělají se implicitní testy na validitu dávky (např. duplicita identifikátorů nebo přítomnost čísla dávky nebo povinných atributů). Ke každému testovanému dokumentu je možné zobrazit

výsledek jeho testování.

6 Konfigurace

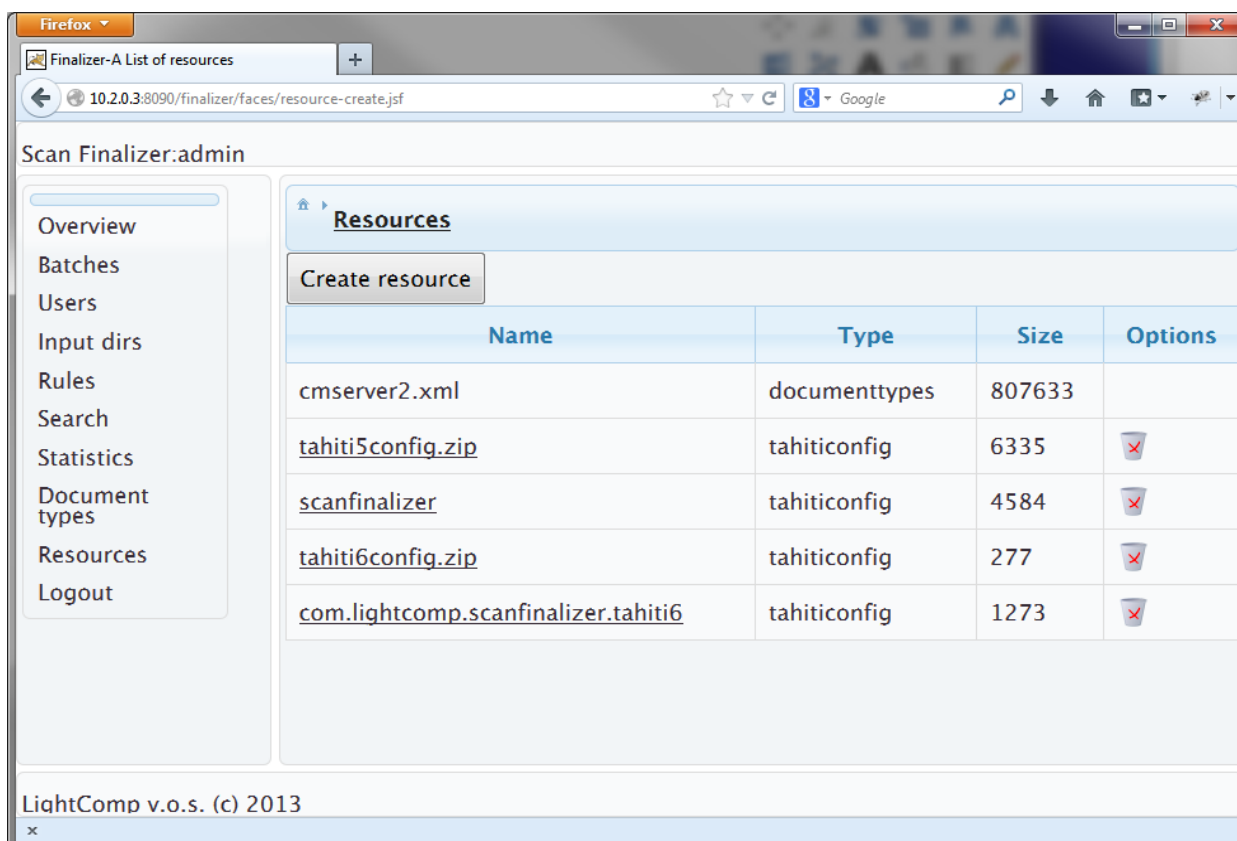
6.1 Tahiti 6

Pro připojení aplikace Tahiti 6 k LightComp ScanFinalizer musí být definovány tyto zdroje:

- `tahiti6config.zip` – konfigurační soubor pro Tahiti
- `com.lightcomp.scanfinalizer.tahiti6` – balíček obsahující formuláře

Postup přidání zdrojů:

1. Přihlášení do aplikace jako administrátor
2. Na stránce Resources vytvořit zdroj `tahiti6config.zip`
 1. Nastavit: Hash type = md5
 2. Nahrát soubor: `tahiti6config.zip`
3. Na stránce Resources vytvořit zdroj `com.lightcomp.scanfinalizer.tahiti6`
 1. Bez udání Hash type
 2. Nahrát soubor `com.lightcomp.scanfinalizer.tahiti6`



Obrázek 1: Nastavení zdrojů pro Tahiti6