# Kontrakt pro integraci systémů v rámci projektu NDK

Následující pravidla vychází ze skutečnosti, že v systému existují minimálně 2 workflow (business, transformační), každé workflow na základě určitých scénářů posunuje stav zpracovávaného objektu.

## Základní pravidla

1. Primárním a jediným referenčním zdrojem dat je obsah pracovního adresáře
2. Pracovní adresář vzniká v okamžiku, kdy je nad daným objektem nutno spolupracovat
3. Pracovní adresář zaniká v okamžiku, kdy se objekt dostane do finálního stavu, který je neměnný
4. V pracovním adresáři jsou uloženy všechny informace potřebné pro rekonstrukci stavu v dalších závislých systémech
5. S pracovním adresářem pracuje vždy právě jedna výkonná jednotka a provádí jednu operaci
6. Každá operace nad pracovním adresářem je idempotentní (může být vykonávána opakovaně se stále stejným výsledkem – zajišťuje způsob implementace operace, řeší problém ztráty výsledkových notifikací)
7. Jednotlivé operace jsou
   1. Mezi objekty (pracovními adresáři) nezávislé
   2. V rámci jednoho objektu z podstaty řízení sekvneční
8. Každá operace nad pracovním adresářem generuje notifikace
   1. Zahájení
   2. Postup (kolik procent hotovo)
   3. Ukončení (úspěch/neúspěch)
   4. Specifické
9. Přístup k pracovnímu adresáři je chráněn přesně definovaným zámkem na úrovni souborového systému
   1. Pro práci se zámkem pracovního adresáře bude k dispozici společný kód ve formě Java knihovny a spustitelného příkazu
      1. Operace získání zámku
      2. Operace potvrzení úspěšného dokončení
      3. Operace chyby a návratu do stavu na začátku
   2. Správa zámků běží přímo na fileserveru
   3. Zámky jsou vytvářeny systémovým voláním pro zamykání na úrovni filesystem (flock)
10. Integrita pracovního adresáře je chráněna verzováním (lokální git)
    1. Lokální .git je chráněn pomocí přístupových oprávnění a běžný proces ho nemůže „rozbít“
    2. Po získání zámku adresáře je zároveň vytvořen branch
    3. Při uvolnění zámku je aktuální stav potvrzen a je proveden merge/commit
    4. Při selhání operace možné
       1. V rozpracované práci pokračovat po restartu
       2. Vrátit pomocí verzování data do stavu před spuštěním operace a začít znovu
11. Komunikace mezi komponentami a předávání řízení
    1. Probíhá ve formě předávání asynchronních zpráv pomocí centrální fronty
    2. Je používán výhradně PUB/SUB režim
    3. Fronta je provozována v HA režimu
    4. Fronta je persistentní
12. Zotavení po výpadku je prováděno na základě dat v pracovním adresáři tzv. recovery službou (dále služba)
    1. Vždy existuje nejvýše jedna instance služby (resp. nad daným pracovním adresářem nikdy nebude soupeřit více instancí)
    2. Služba prochází pracovní adresáře (inicializace bude upřesněna)
       1. V běžném provozu
          1. timeout, když je zámek nad adresářem držen dlouho - generuje task pro manuální obsluhu
          2. manuálně nad konkrétním adresářem
       2. Při startu systému v režimu recovery, kdy automaticky vrací rozpracované úlohy do počátečního stavu a notifikací umožní jejich restart
    3. Služba uvede adresář do stavu vhodného k restartu operace (rollback změn)
    4. Služba na základě údaje v pracovním adresáři generuje notifikace, pomocí kterých ostatní systémy (workflow, transformace) aktualizují svůj stav
13. Zapnutí systému
    1. Nastartuje se fronta
    2. Nastartují se procesní uzly
    3. Spustí se recovery
14. Vypnutí systému
    1. Ukončí se fronta
       1. Varianta – ukončí se přijímání požadavků na úlohy měnící stav objektů a počká se na dokončení zpracování
       2. Aktuálně čekající zprávy jsou zapomenuty
    2. Násilně se ukončí procesní uzly