

Implementation of workflows as Finite State Machines in a national doctoral dissertations archive

Nikos Houssos, Dimitris Zavaliadis,
Kostas Stamatis and Panagiotis Stathopoulos

National Documentation Centre / National Hellenic Research Foundation, Greece



Outline

- Introduction to Hellenic National Archive of Doctoral Dissertations
- HEDI workflow
- Workflow modelling and implementation as an FSM engine
- The FSM engine design and implementation
- Summary – conclusions – future work

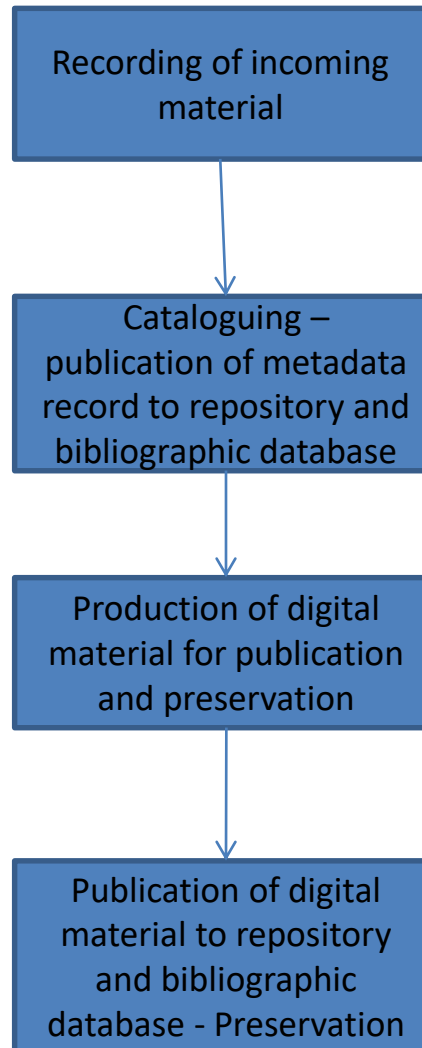
Introduction to HEDI

- Hellenic National Archive of Doctoral Dissertations
 - ~25070 theses, ~3 million pages (September 2011)
 - ~1400 dissertations arriving every year
- Until recently
 - Cataloguing using the library automation system ABEKT
 - Public web access through the ARGO bibliographic database (based on Z39.50) and an online reader

HEDI – new IT infrastructure

- e-theses repository on DSpace -> user friendly interface for searching and browsing
- New application for the support of EKT internal workflows for HEDI (cataloguing, quality control, processing of digital material)
- Support of ARGO continues
- Infrastructure based entirely on open source software platforms and components

HEDI workflow - overview



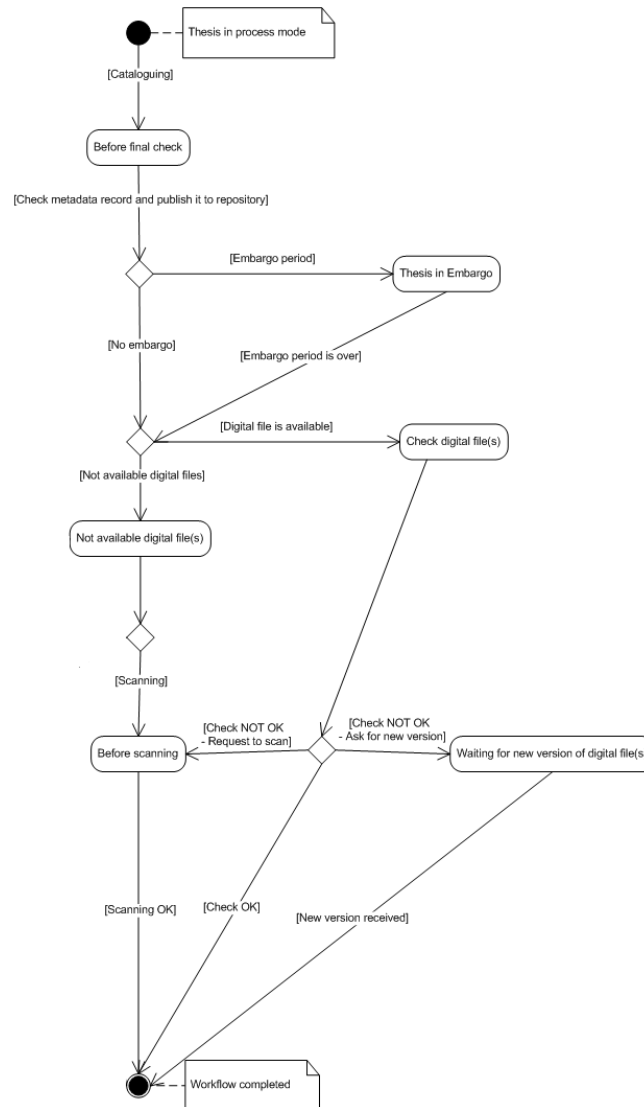
HEDI workflow

- Four major steps:
 - Recording of incoming material – one record per dissertation, much shorter than the full metadata record
 - Cataloguing – results in a full metadata record in UNIMARC. Publication of record to HEDI and ARGO.
 - Processing of digital material (e.g. OCR, format transformations)
 - Publication of dissertation full-text to both the HEDI repository and ARGO, availability for online reading via an open source reader

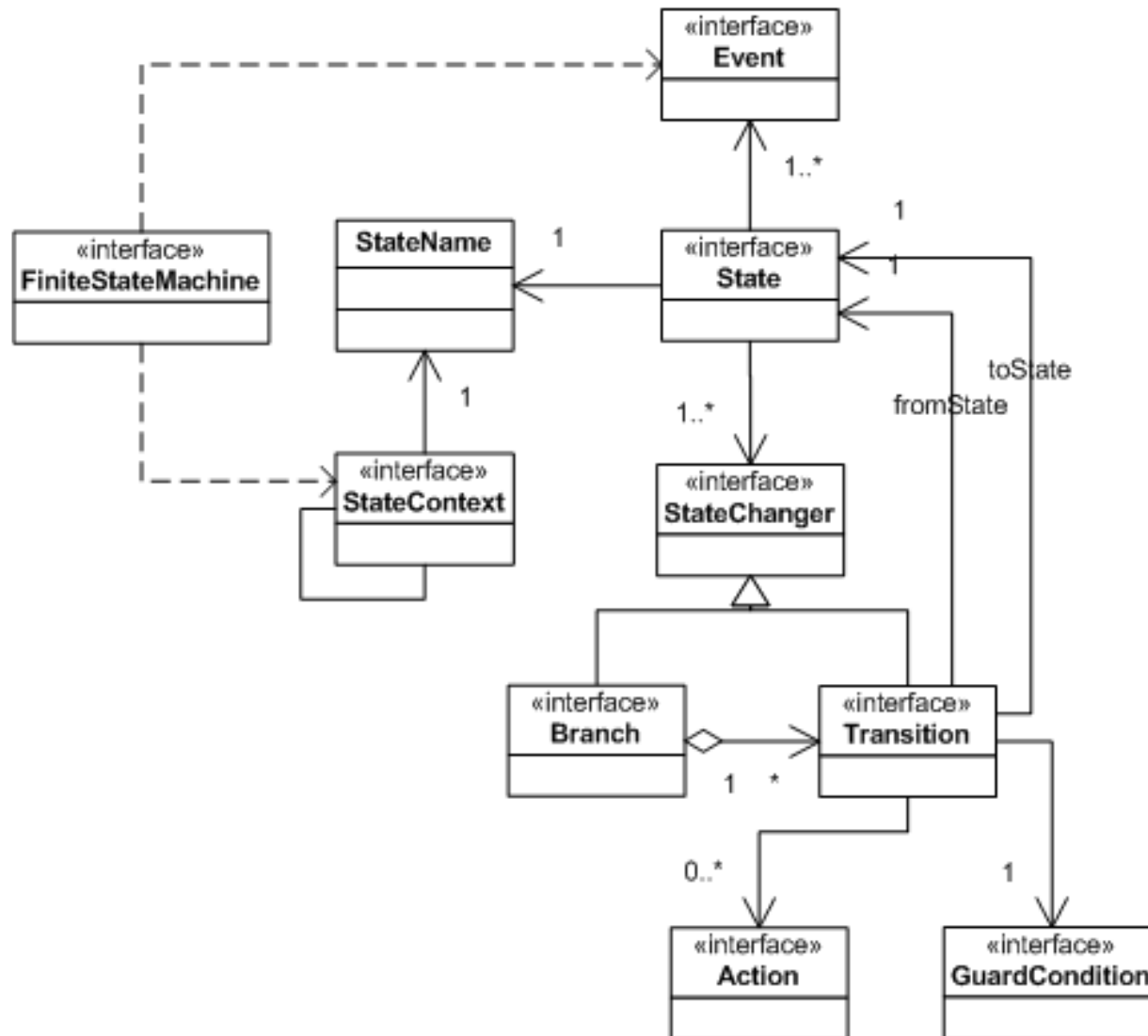
HEDI workflow implementation

- Workflow modelled as a Finite State Machine (FSM)
- Specification of workflow in a formal manner
- Workflow implemented (in Java) through a generic Finite State Machine library developed by EKT
- Branches, guard conditions supported
- Workflow definition in an XML configuration file – easy to modify outside source code
- FSM engine independent of DSpace, re-usable in other systems

HEDI workflow as an FSM



FSM engine domain model



Summary - Future work

- Modelling and implementation of HEDI workflows as Finite State Machines
- Generic, re-usable Finite State Machines library implemented in Java – easy to define and modify workflows.
- Future work:
 - Support online submission to HEDI by universities and doctoral graduates, more complex workflows
 - Publish the FSM engine as open source software

Thank you!

- More info:

nhoussos AT ekt.gr