Editor 3.0: Redesigning the YAWL User Interface

Michael Adams

Queensland University of Technology
Brisbane, Australia
mj.adams@qut.edu.au

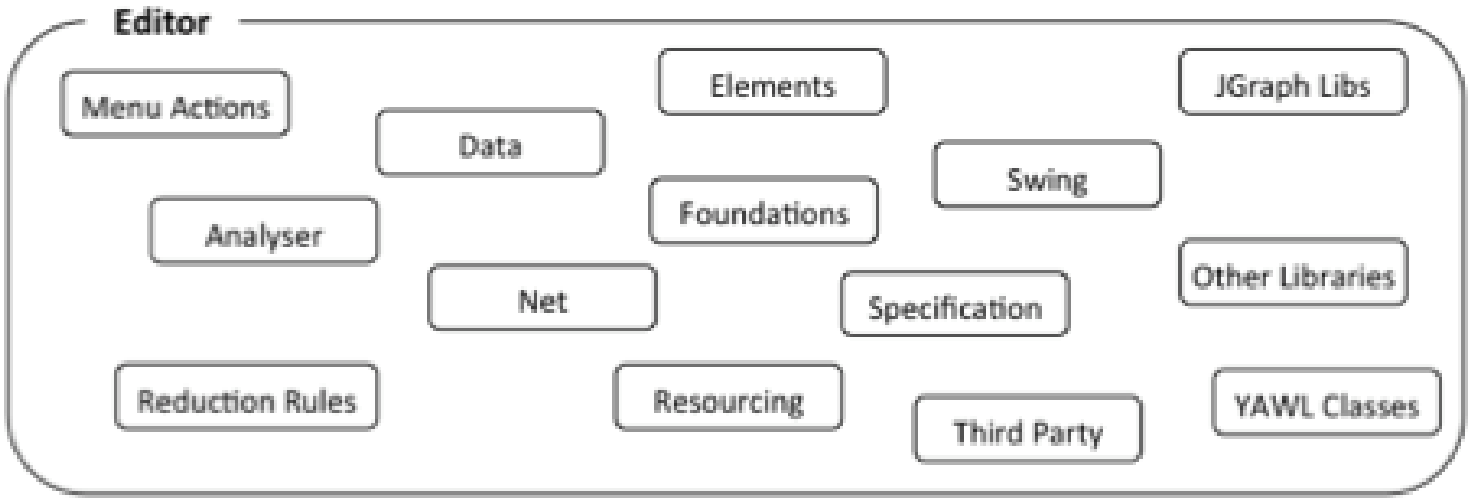
YAWL Symposium, Sankt-Augustin, Germany, 7 June 2013

Legacy YAWL Editor

- The Editor is the ‘public face’ of YAWL System
- Almost a decade old
  - UI essentially unchanged
- Prototypical base
  - Never ‘thrown away’!
  - Originally control-flow only
  - Later grafted on data and resource perspectives, timers, configuration, extended attributes, …
- Monolithic architecture
  - Difficult to maintain and enhance

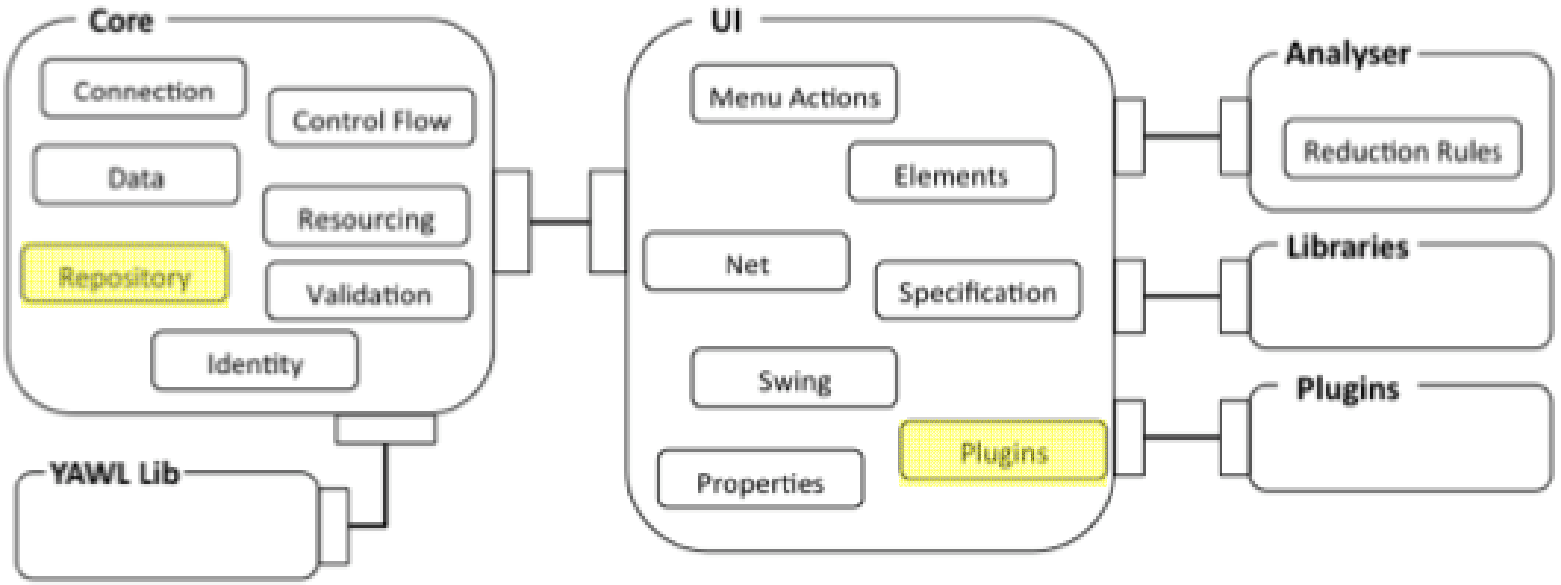
Legacy Architecture

- Flattened package structure
- Mixes UI, Data and Transport
- Much duplication of existing functionality
- Redundant and error prone



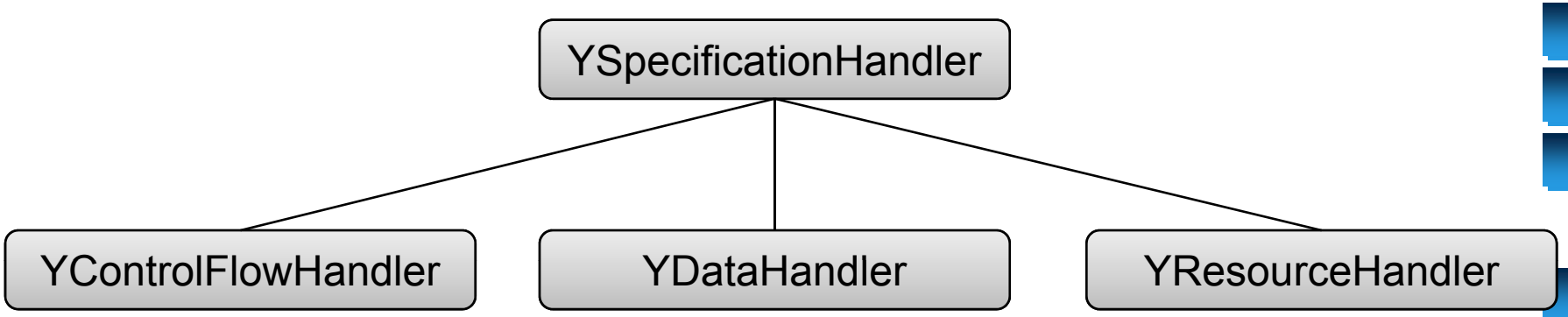
Editor 3.0 Architecture

- Complete separation of core and UI
- Core library with concise API
  - Handles all interactions with engine and services
  - Much faster file opens and saves
- Any number of UIs supported



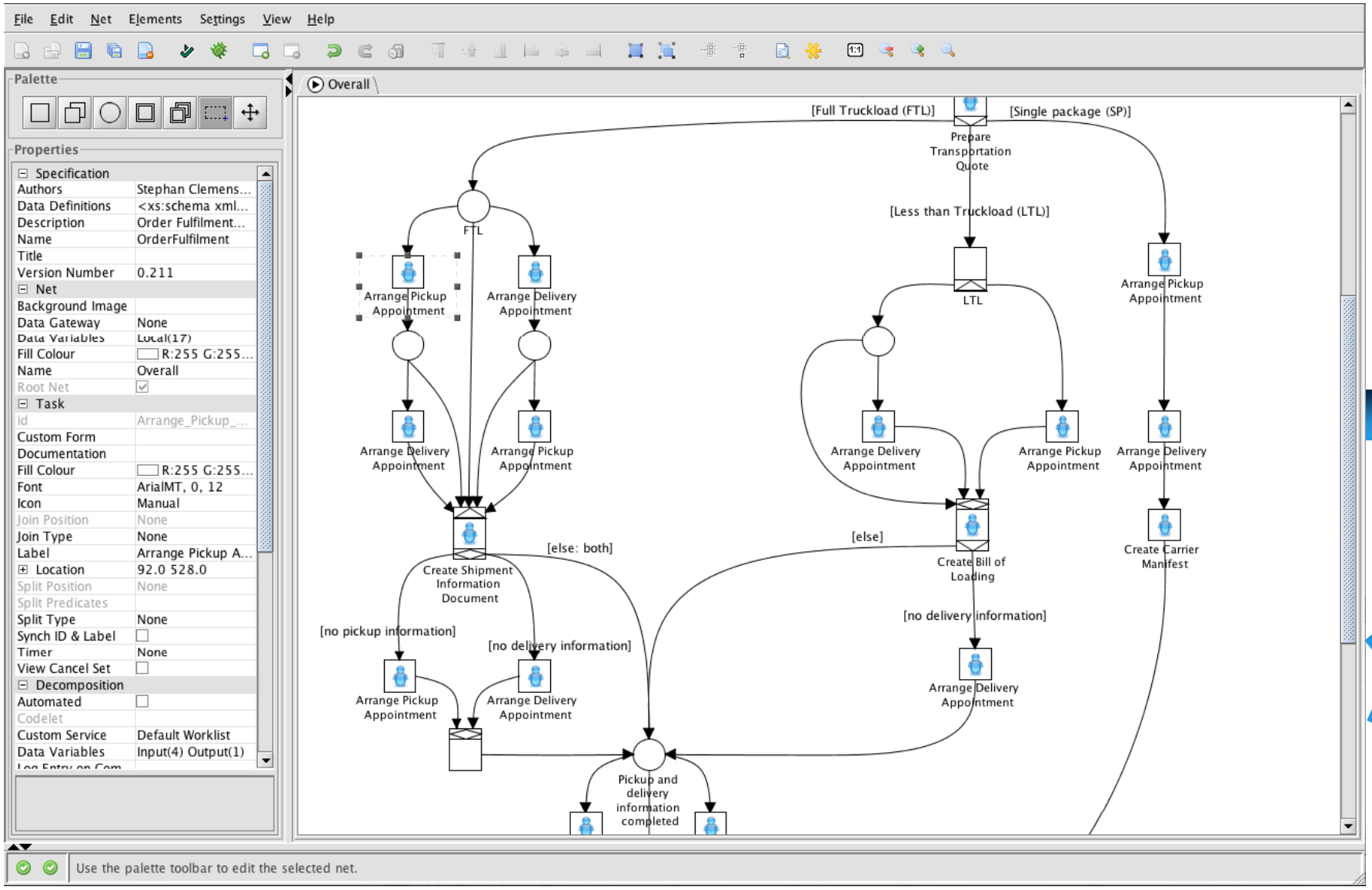
Core Library

- Allows handling of multiple concurrent specs



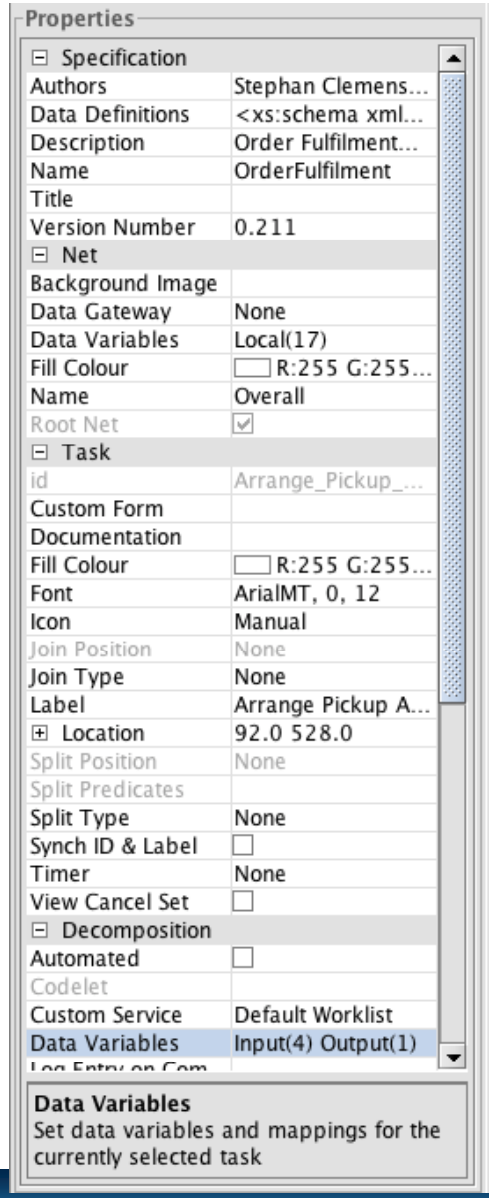
- Supports a repository for saving and reusing:
  - Net Decompositions: parent- and sub-processes, may be interchanged
  - Task Decompositions
  - Data Type Definitions: entire or sub-sets
  - Extended Attributes

User Interface



Properties Window

- All properties now co-located
  - Greatly improves ease of use
  - Replaces many scattered dialogs
- Continuous access to specification, net, task and decomposition levels
- Includes data, splits & joins, icons, timers, etc.
- Values shown in ‘native’ formats



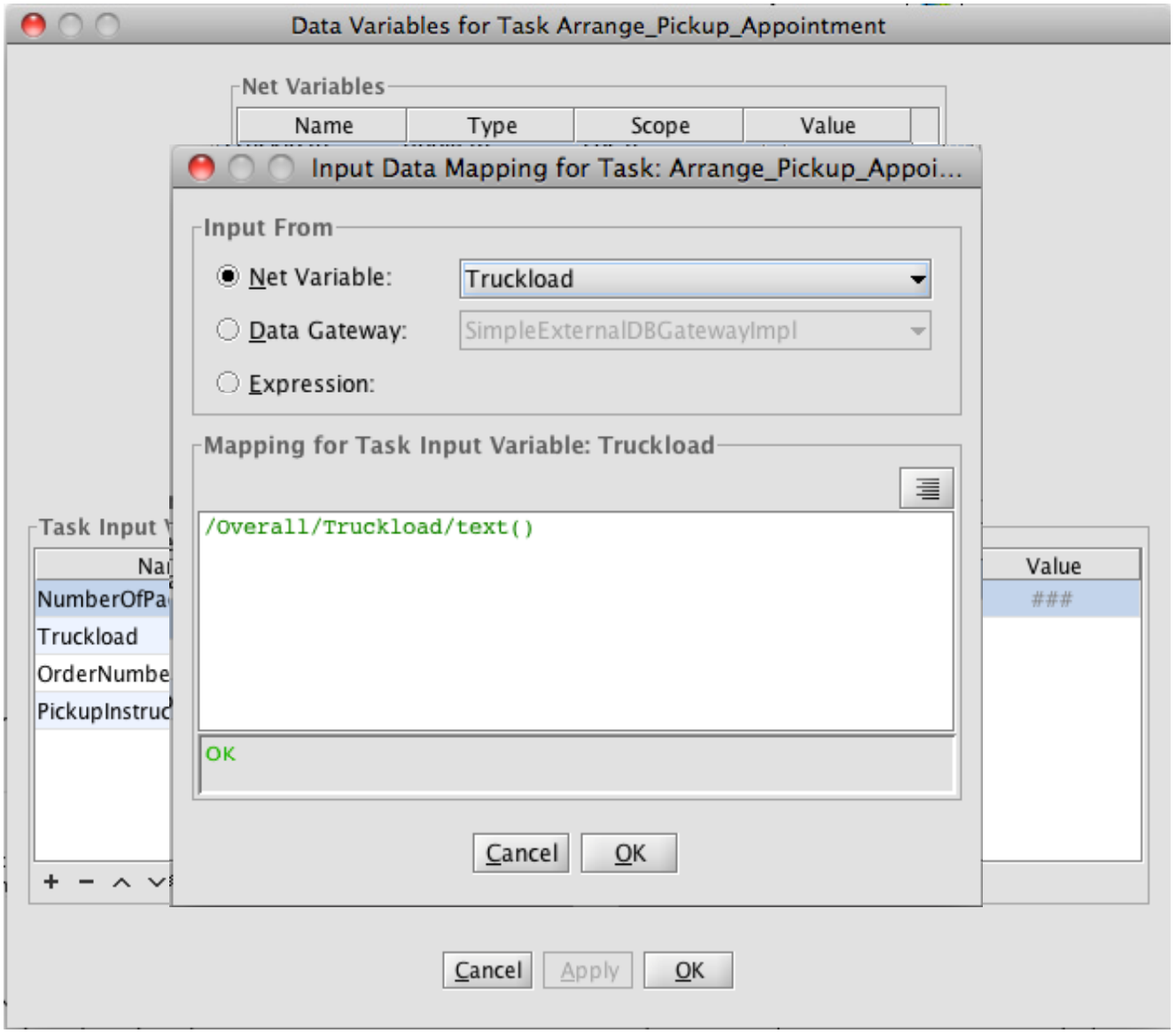
Data Perspective

- Arguably the most difficult aspect for novice (and not-so-novice) users:
  - XPath & XQuery
  - XML Schema
  - Net variables & initial values
  - Task decomposition variables & default values
  - Variable Types: Input, Output, Input & Output, Local
  - Mappings
  - Multiple Instance Tasks: accessor, splitter, instance & aggregator queries

Data Perspective

- New single dialog for task and net-level variables
- Confusing Input & Output notion is gone
- Mappings automatically generated through drag and drop
  - Even for multiple instance tasks!

Data Dialog


Data Variables for Task Arrange_Pickup_Appointment
Net Variables
Name
Type
Scope
Value
Input Data Mapping for Task: Arrange_Pickup_Appoi...
Input From
Net Variable:
Truckload
Data Gateway:
SimpleExternalDBGatewayImpl
Expression:
Mapping for Task Input Variable: Truckload
/Overall/Truckload/text()
OK
Cancel
OK
Task Input
NumberOfPa
Truckload
OrderNumbe
PickupInstruc
Value
###
Cancel
Apply
OK

Easier Data Type Definitions

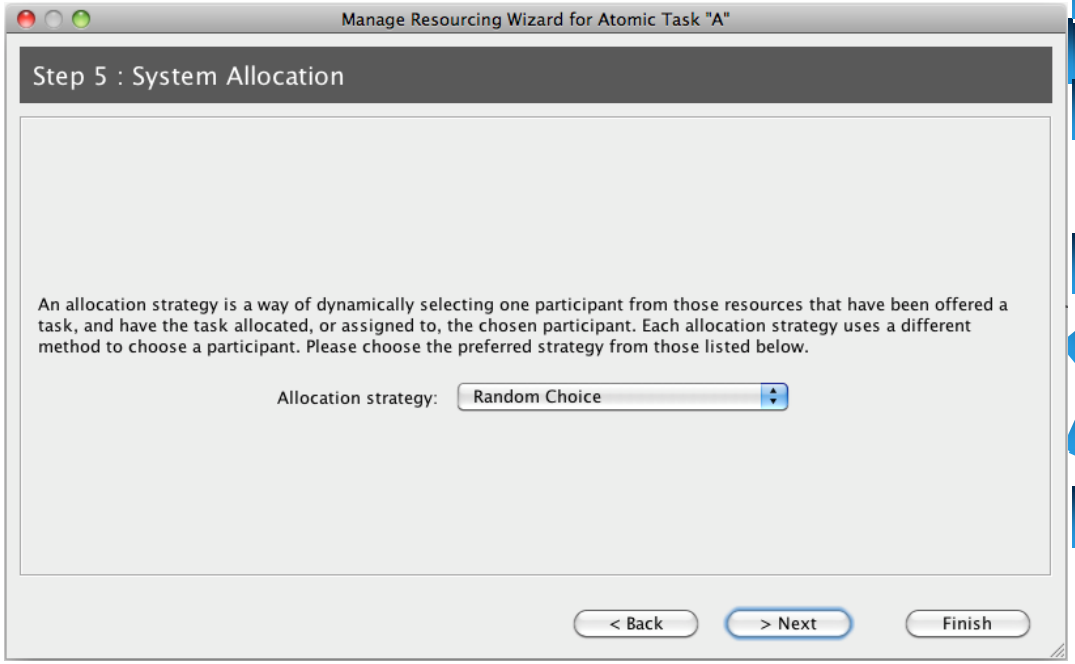
- Simplified data definition language
  - Resembles *Pascal*
- User-switchable between views:

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="BookOrder">
    <xs:sequence>
      <xs:element name="title" type="xs:string"/>
      <xs:element name="price" type="xs:double"/>
      <xs:element name="inStock" type="xs:boolean"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="BookList">
    <xs:sequence>
      <xs:element maxOccurs="unbounded" name="order" type="BookOrder"/>
    </xs:sequence>
  </xs:complexType>
</xs:schema>
```

```
type BookOrder = record {
    title: string;
    price: double;
    inStock: boolean;
}
type BookList = order(1..n) of BookOrder;
```

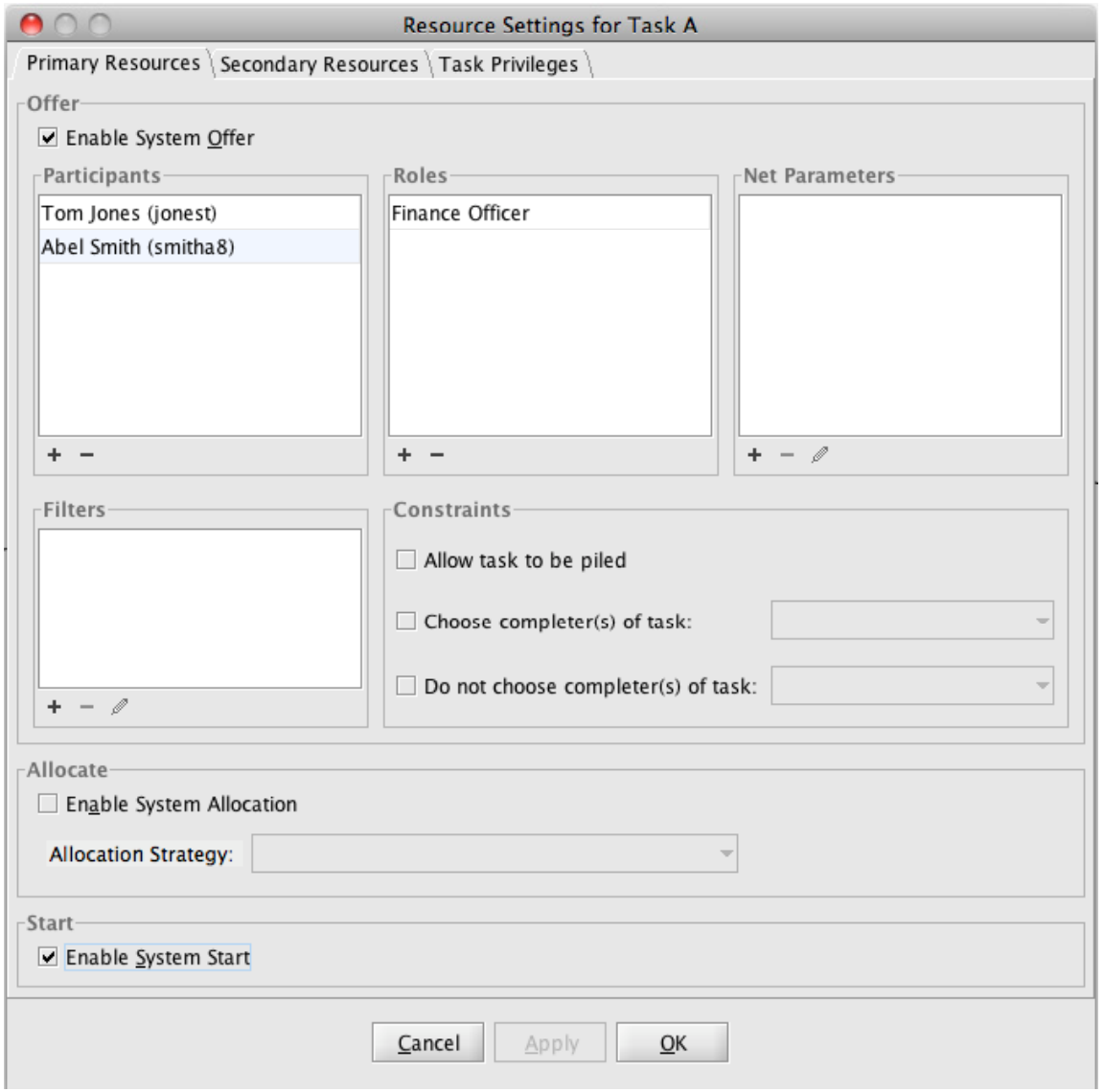
Legacy Resource Perspective

- Wizard metaphor:
  - dated
  - Forced navigation from page 1
  - Immediate save – no undo or cancel
  - Always on top
  - Too verbose



Resource Perspective

- All settings in one place
- Only shows added resources
- Easy to access any aspect
- Can cancel before save
- Swim lanes will also be added to the canvas



Conclusion

- Outlined the major new features of the YAWL Editor
  - Almost every aspect has been improved
- Envisaged outcomes:
  - Easier use → more positive user experience
  - Easier to teach, since novice users no longer face steep learning curve (esp. data perspective)
- Other possible functional additions:
  - 'Direct Service' task decompositions
  - 'Perspective overlays'
  - Simulation
  - Any other ideas?
- An alpha version is expected to be made available to testers and early adopters in August