

Backend, Service, Listener
VuFind's new SOLR connection

Originally Presented By

David Maus
Herzog August Bibliothek Wolfenbüttel

VuFind Anwendertreffen
September 16th - 17th, Hamburg Harburg

VuFind 2.x Paradigms

- **Based on Zend Framework 2**
 - Hint: If in doubt, check the framework's source code
- **PHP Namespaces and ClassLoader**
 - Search term: PSR-0
- **Front Controller Pattern**
 - Make decisions up-front, then execute
- **ServiceManager as central registry**
 - ...not perfect, but better than global variables

Problems with the old SOLR connection

- Organic growth (PHP4, PHP5)
 - "works for me"
- Tight coupling of functionality ("talk to SOLR"), index schema, and web application
- One class does everything
- Hard to extend programmatically
 - ...except "works for me"

VuFind 2.x SOLR connection

➤ Separation of concerns

Processing a user query implemented by exchangeable objects

Separation of functionality and application

- VuFindSearch – Functionality ("Framework")
- VuFind\Search – Specialized instances of framework classes ("Application")

➤ Parameter aggregation

Search parameters are aggregated from different sources

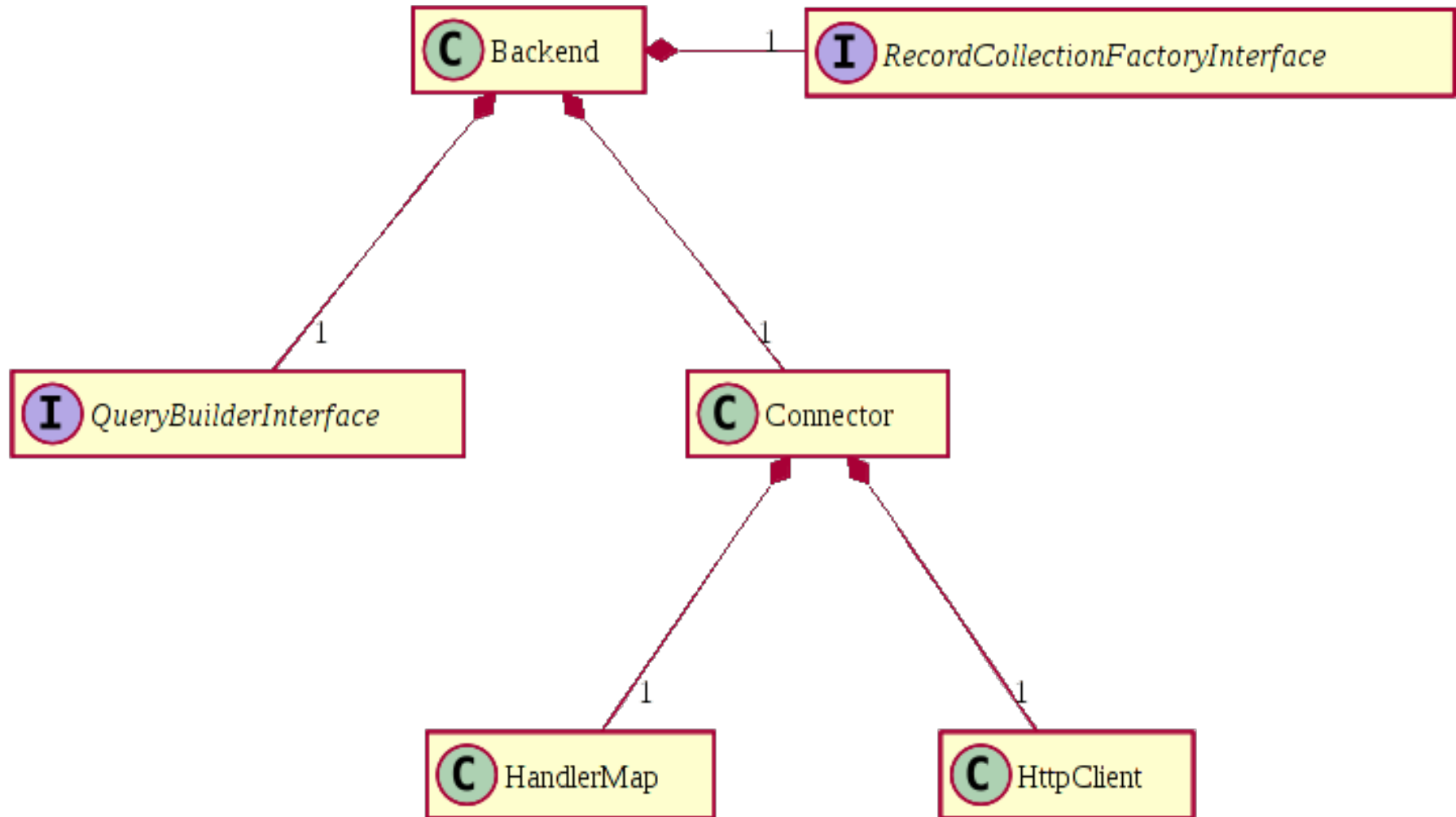
Close match of SOLR-Functions and VuFind-Classes

➤ Events as extension points

Flexible EventManager provided by the Zend Framework

Modification of search behaviour by custom modules is possible

VuFind 2.x SOLR Backend



VuFindSearch\Backend\Solr\Backend

- Translates search requests into SOLR parameters
- Defines the response writer (hardcoded)
 - `wt=json&json.nl=arrarr`
- Uses QueryBuilder to translate search (field, term) in SOLR search syntax
 - includes `spellcheck.q` and `hl.q`
- Delegates to Connector class
- Deserializes the response
- Creates response using the factory pattern
 - `VuFindSearch\Backend\Solr\Response\Json\RecordCollectionFactory`

VuFindSearch\Backend\Solr\Connector

- Connection to SOLR index
- Uses HandlerMap to map search operations (search, retrieve, morelikethis, ...) to SOLR Request Handlers
 - VuFindSearch\Backend\Solr\HandlerMap
 - Supports Appends, Defaults, Invariants analogous to SOLR
- HTTP Transport
- Error handling: Triggers an exception in case of error
 - VuFindSearch\Backend\Exception\BackendException
 - VuFindSearch\Backend\Exception\RemoteErrorException (HTTP 5xx)
 - VuFindSearch\Backend\Exception\RequestErrorException (HTTP 4xx)

VuFindSearch\Response\RecordCollectionFactoryInterface

- Executing a search operation always returns a RecordCollection
 - ! Support for other response types (e.g. terms, alphabetic browse) would be nice
- RecordCollectionFactory creates RecordCollection based on the SOLR response
 - ! Hard-coded response writer, only JSON is currently supported
- RecordCollectionFactory instantiates appropriate RecordDriver objects
- RecordCollection acts as wrapper for the entire SOLR response (Facets)

VuFind\Search\BackendManager

- Backend configuration
- Uses a local ("scoped") ServiceManager to create backend instances
- Backend instances are addressed by a distinct name
 - Aliases can be used and are helpful in some circumstances
- Factory Pattern to instantiate and configure backends

VuFindSearch\Service

- Integrates backends and backend operations
- Backend instances are addressed by name
- Uses the event system to resolve a backend name to a backend instance
- Executes events before (pre) and after (post) a search operation
- Executes event in case an error occurred (error)
- Defined objects (listeners) can react to the events

Event: VuFindSearch\Service::EVENT_RESOLVE

- "short-circuited": Stops as soon as a listener provides a backend
- Event contains information about the search operation ("context") and all formal parameters
- Association between backend name and backend instance is cached internally

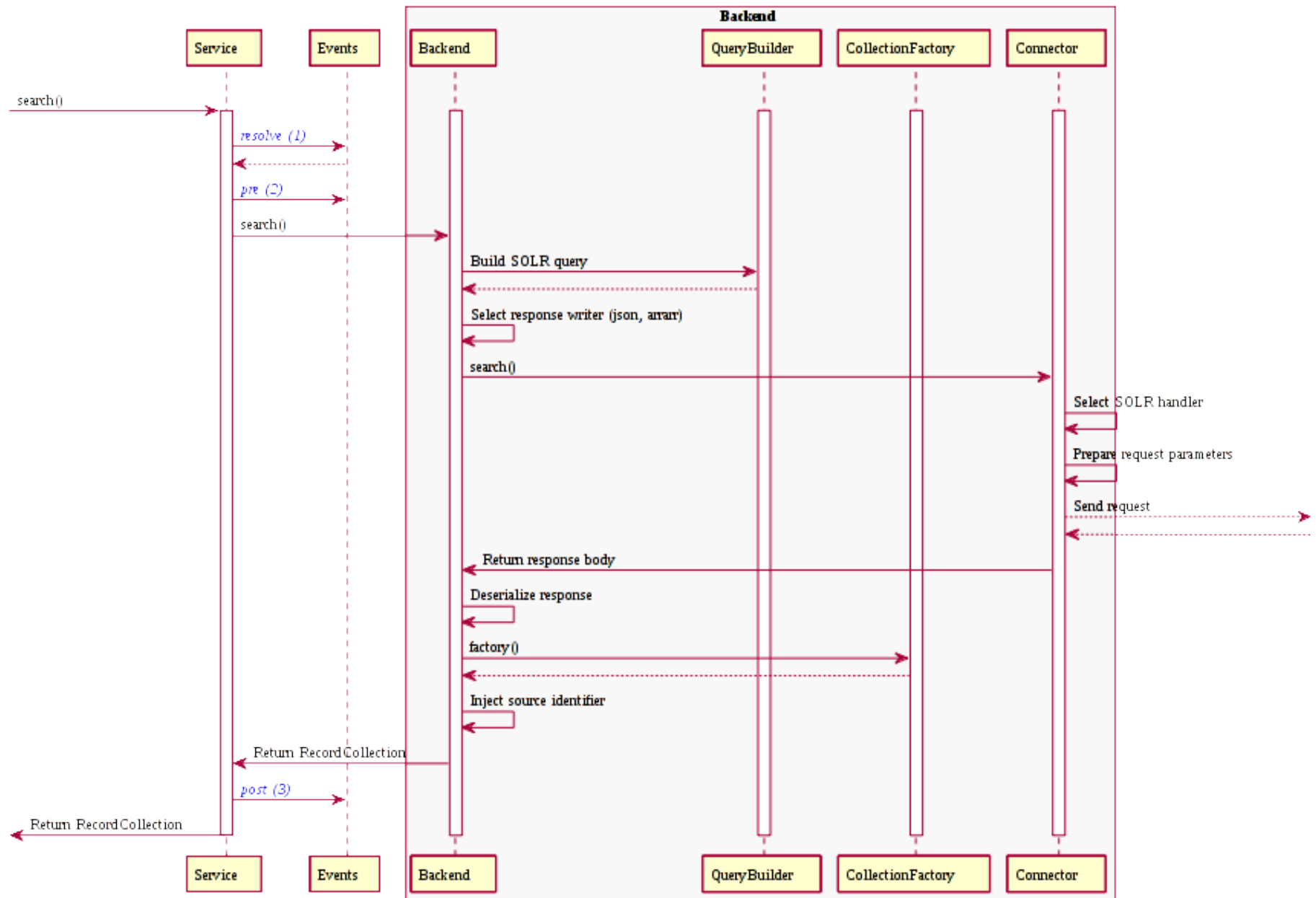
Event: VuFindSearch\Service::EVENT_PRE/EVENT_POST

- EVENT_PRE contains the resolved backend, name of operation and all formal parameters
- EVENT_POST contains the search response (RecordCollection)
- Listeners can manipulate the response and secondary search parameters

Event: VuFindSearch\Service::EVENT_ERROR

- Service catches exceptions from the backend
- The error event contains the exception
- Exception is rethrown
- Use-case: Support SOLR 3.x and SOLR 4.x error responses
Response bodies are different (text/plain with stack trace vs. real JSON response)

Herzog August Bibliothek Wolfenbüttel



VuFind 2.2 (Summer 2014)

- More parameter aggregation at the higher levels
 - Options – Params – Results (SearchObject)
- Custom faceting behaviour?
 - Your feedback!**
- Support other response types
 - Terms/AlphabeticBrowse => Systematic Access ftw!