Node View Access API version 2.0

Piotr Poznański

NVA-API Overview (1)

- New NVA-API is based on version 1.0 – extension and minor changes of the API core,
- Adapted to new futures of the Pan language,
- Some features depend on LLD-XML and CCM,

INTACT, ADDED, MODIFIED, REMOVED

NVA-API Overview (2)

- The core API:
  - Configuration, locking and notifications handling (`CacheManager, Configuration`),
  - Path manipulation (`Path`),
  - Data querying (`Element, Property, Resource, List, NList`),
  - Metadata querying (`Metadata`).
- API extension to facilitate the usage of API (extend: `Configuration, Element, Path`).

Property Interface (1)

- Property interface (reflects Pan data types)
  - `string getStringValue()`
  - `boolean getBooleanValue()`
  - `long getLongValue()`
  - `double getDoubleValue()`

`string` can hold arbitrary data (like in Perl, and Java `String`, not just `char*`)

(`int getIntValue()` and `float getFloatValue()` removed)

Property Interface (2)

- `BLOBBuffer` interface disappears,
- redefinition of types:

BLOB \ remote	**YES (ref. by URL)**	**NO (in the profile)**
YES (`open,get,close`)	`stream`	`embed`
NO (`getStringValue`)	`fetch`	`string`

Property Interface (3)

- `link` is dereferenced automatically by the API, and visible as element that it points to (however we may add a call to get the link path)

Resource Interface

- Every resource may be still queried for children in the iterator-like (old) way:
  - `boolean hasNextElement()`
  - `Element getNextElement()`
  - `Element currentElement()`
  - `ElementName currentElementName()`
  - `void reset()`

List, Nlist Interfaces

- Two additional interfaces:
  - **`List`** – Pan's **`list`** type, list in a given language of implementation, in Perl - list, in Java - **`java.util.List`**,
  - **`NList`** – the **`nlist`** type, Perl – hash, Java – **`java.util.Map`**,
- distinction on **`nlist`** into **`table`** and **`record`** not made (stub generation required, C/C++ **`struct`** or Java classes out of the HLD Schema).

Metadata Interface

- Metadata interface contains:
  - `Type getType()`
  - `MD5Digest getMD5Digest()`
  - `Derivation getDerivation()`
  - `ACL getACL()(?)`
- removed:
  - `Encoding getEncoding()`
  - `Issue getIssue()`
  - `TimeStamp getTimeStamp()`

Summary

- New NVA API is an extension of the version 1.0,
- It reflects the features of Pan,
- It depends on the “CCM: to be or not to be” question and on the LLD-XML (Node View Profile) Specification.