JMC Agent Probe Definition Format

The following things would have to be part of the specification for a probe:

- Data identifying the method to instrument (FQN for the class, method and descriptor)
- Data identifying the parameters and/or return value to record as attributes
    - Parameters/Return values should be able to appear multiple times, with different access paths/converters
    - For following references in parameters/return values (via fields), we probably need to specify the type of the thing referenced in the end
    - May want to consider following references in the object being instrumented too (fields)
- Recording metadata (optional, should be able to generate "ok" defaults)
    - A name (label) (will not look very nice without, but perhaps take class name, and split along camel case boundaries as default?)
    - A description
    - enabled - should the event be enabled by default?
    - stacktrace - capture stack trace by default?
    - category - used to be path, but OpenJDK API calls this category

The following things could be part of the specification for a probe:

- Location (ENTRY, EXIT, WRAP)
  Emits instant event at entry, at exit, or wraps the method body with a timed event. May also want a BOTH (instant event both at entry and exit).
  Default should be WRAP.

Things to remove from the current prototype:

- reuse (use case is simply not there in JDK 11+)
- loggingtype (text logging should be killed)
- Kind (INSTANT, DURATION, TIMED) - Rather OracleJDK specific - skip this
- thread - should the event record thread data - This can be skipped and assumed to always be true for the Oracle JDK
- path Should be switched to Category

Open questions:

1. The prototype currently uses parameter index to identify parameters, with -1 meaning return value. We probably want a separate returnValue element for return values.
2. If we want to add fields or reference chains through fields, then we may want to add a new element "field".