

# Extending XQuery

---

## with Collections, Indexes, and Integrity Constraints

C. Andrei, **M. Brantner**, D. Florescu,  
D. Graf, D. Kossmann, M. Zaharioudakis



**ETH**

Eidgenössische Technische Hochschule Zürich  
Swiss Federal Institute of Technology Zurich

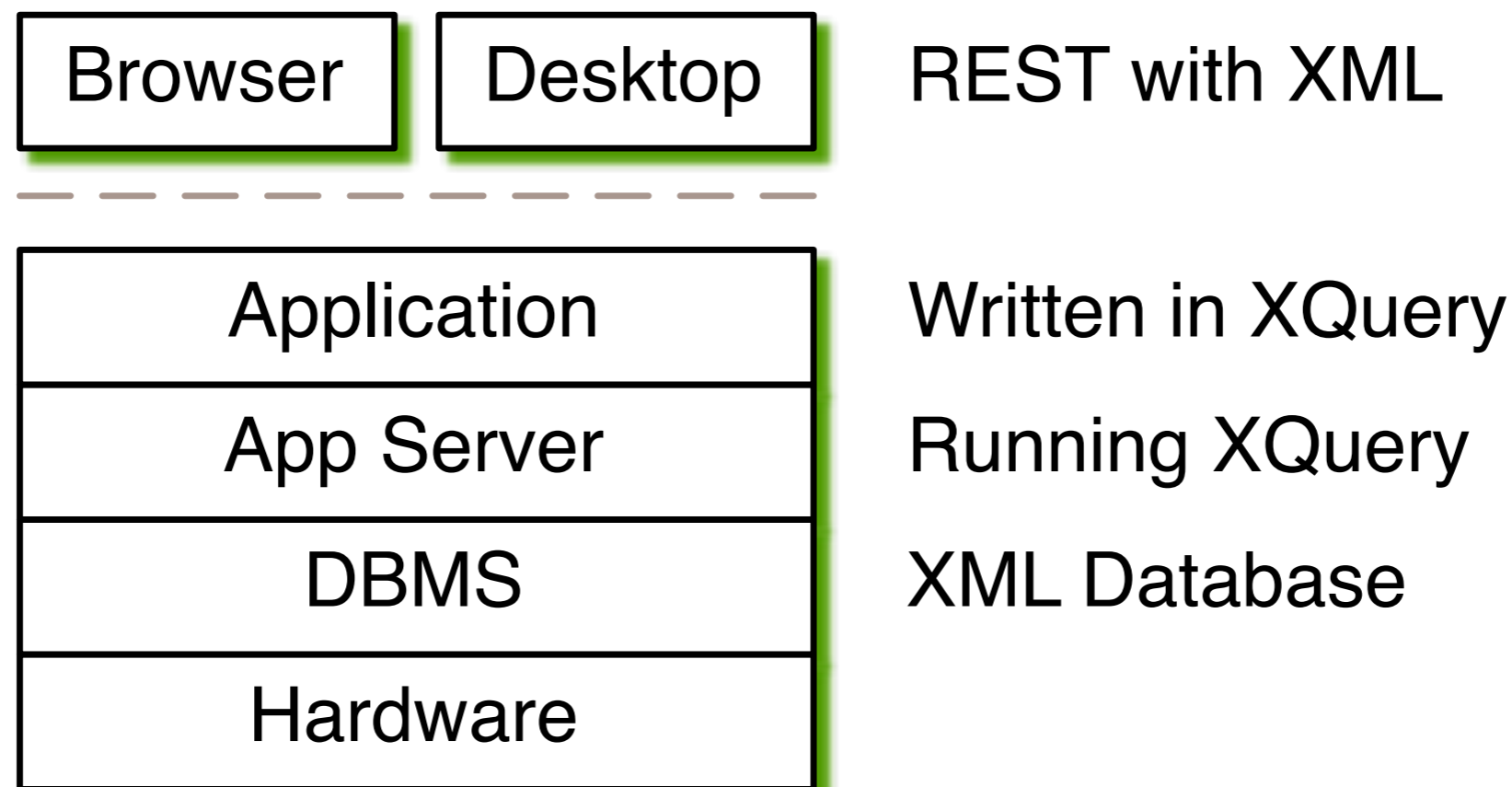
**28msec**

**ORACLE®**

## Context

---

- XQuery - general purpose XML processing language
- temporary information (OO-XQ-OO)
- XML end-to-end architecture



**Problem:** XQuery is incomplete

- model, describe, and reason about the database
- semantics of collections (e.g. copy or order)
- declare and manage
  - collections, indexes, integrity constraints (ICs)

**Goal:** XQuery Data Definition Facility

- bring collections, indexes, and ICs to XQuery
- Extension to XQuery 1.1 & XQuery Update

# Approach & Outline

---

- Extensions to
  - static- & dynamic context
  - processing model
  - modules & prolog
- New function libraries to
  - manage collections, indexes, and ics
  - introspect the static- and dynamic context

# Extensions to the Processing Model

---

Module A

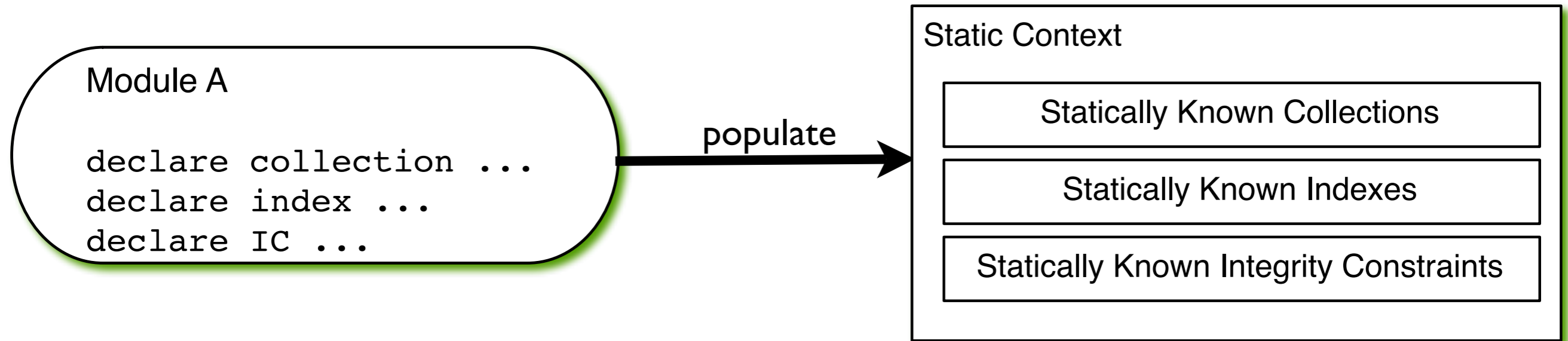
```
declare collection ...
```

```
declare index ...
```

```
declare IC ...
```

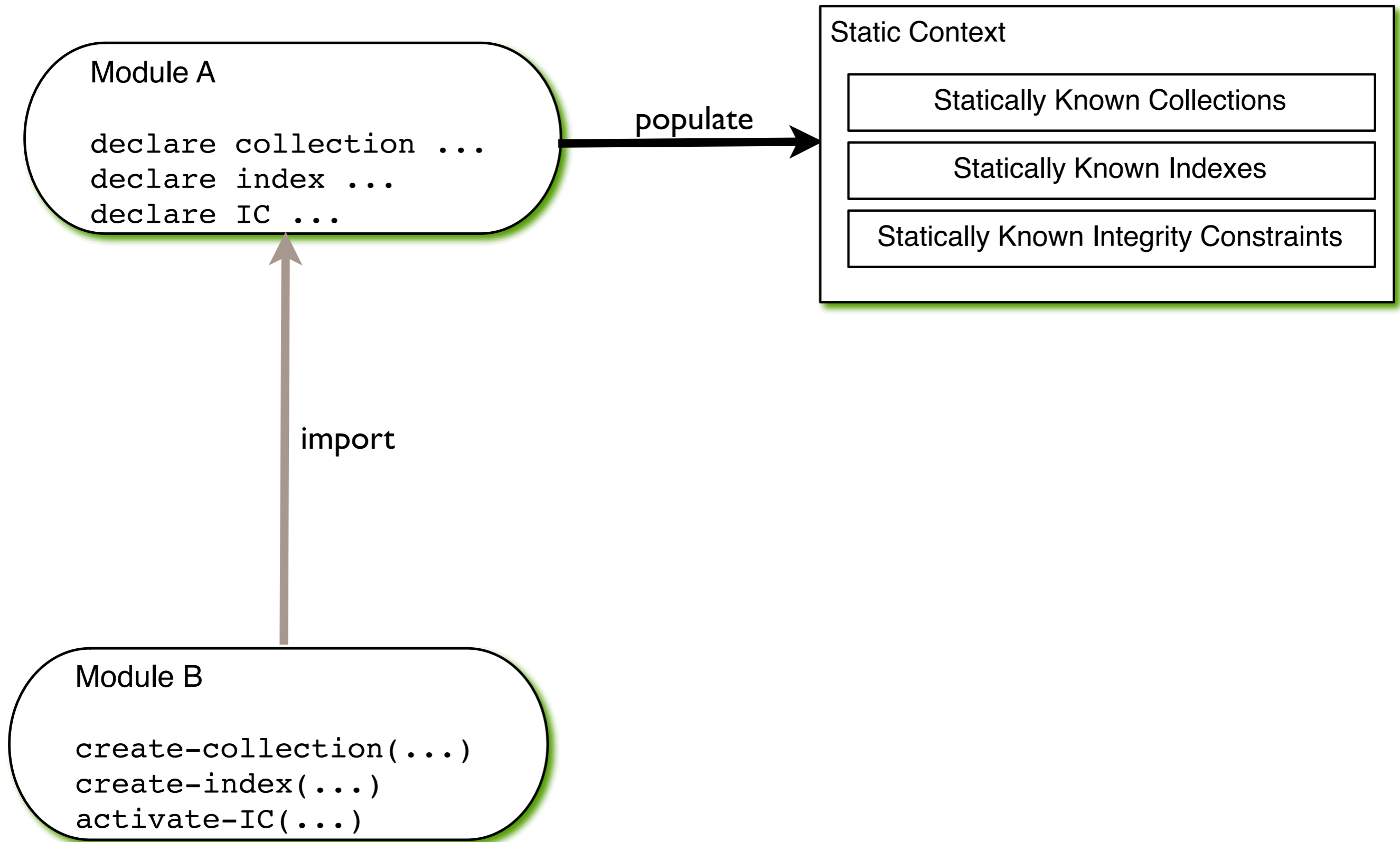
# Extensions to the Processing Model

---



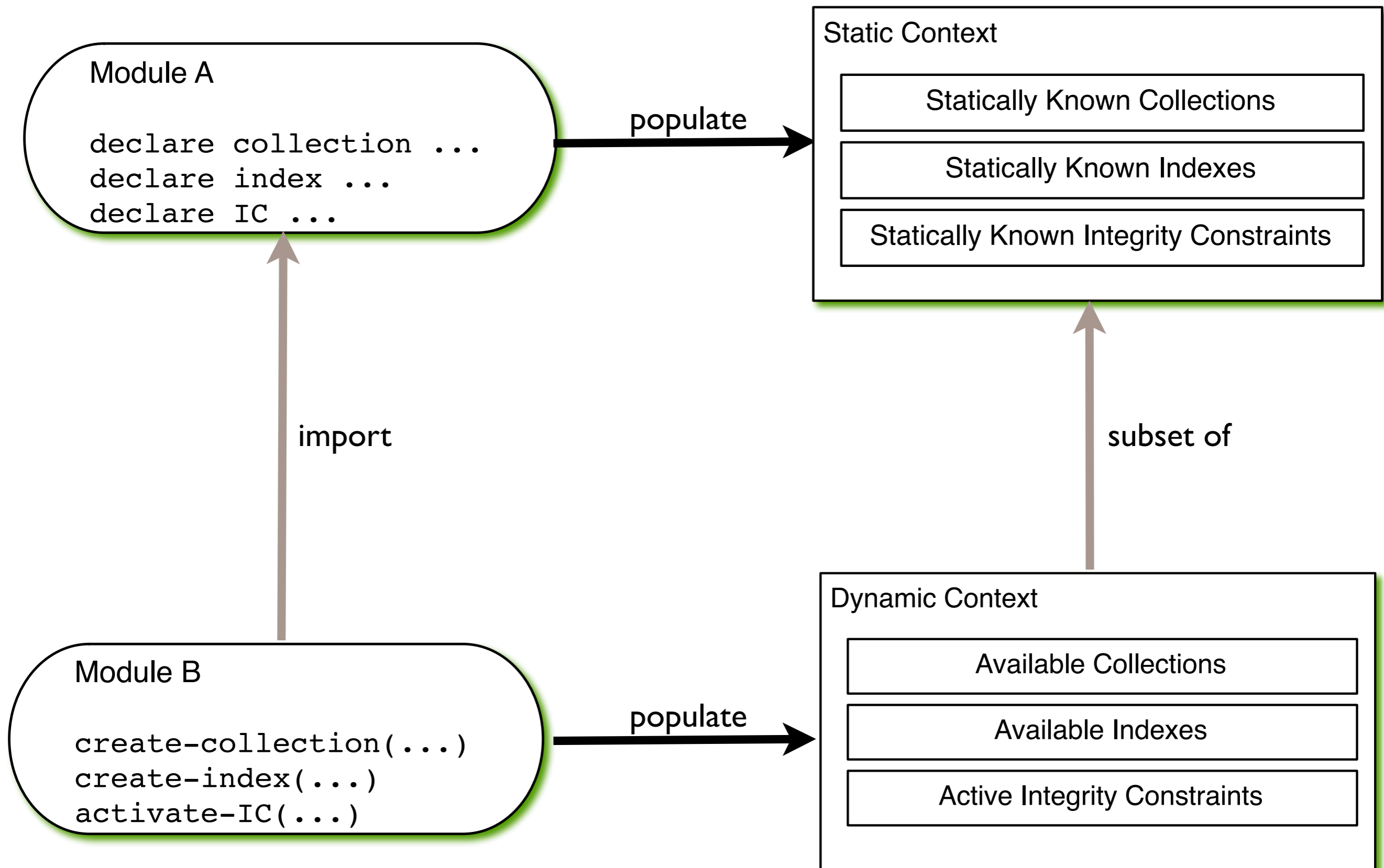
# Extensions to the Processing Model

---



# Extensions to the Processing Model

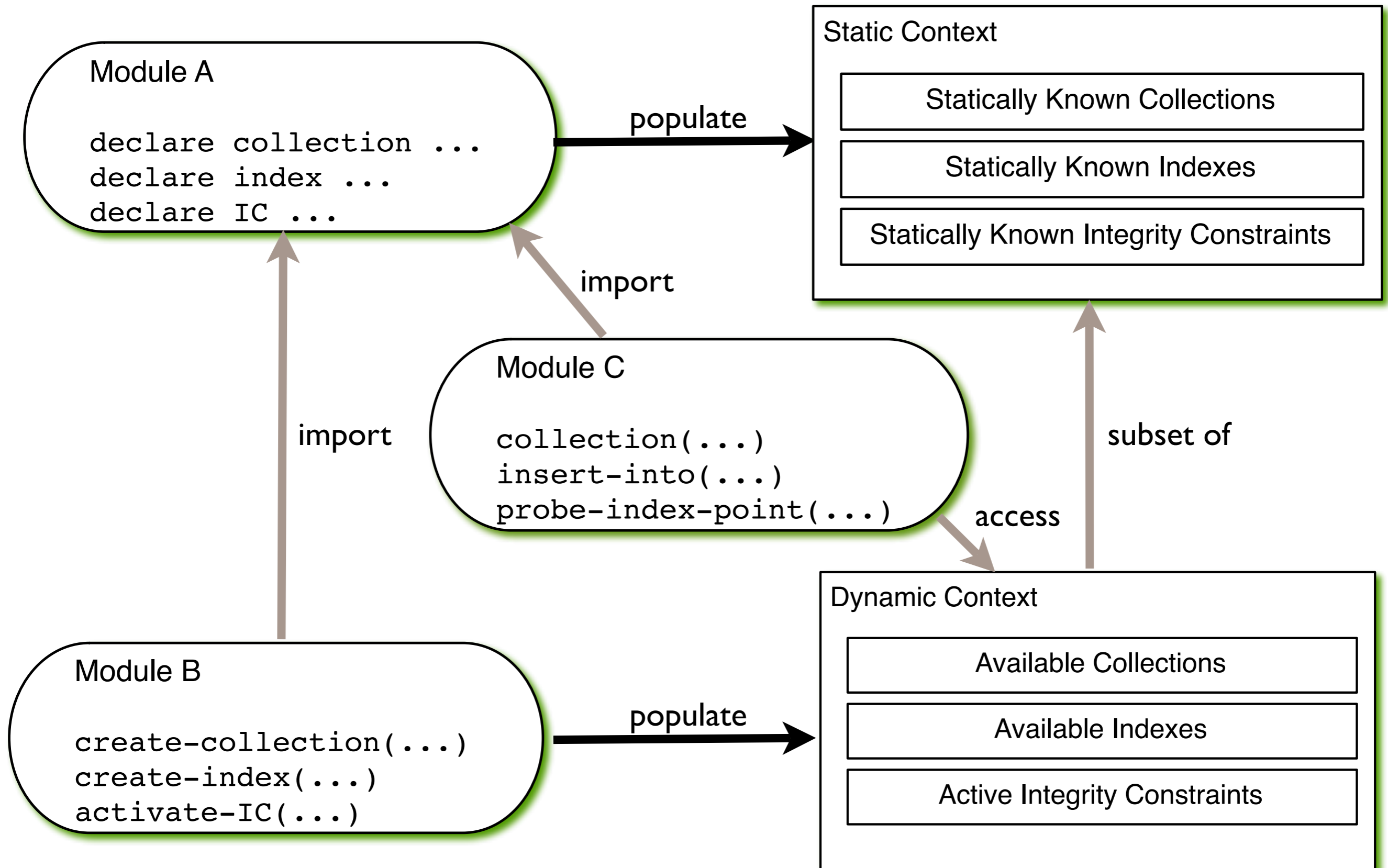
---





# Extensions to the Processing Model

---



# Collections

---

- disjoint sequences of parent-less nodes
- metadata
  - types
  - properties (e.g., ordered, mutable)
  - node modifiers (const or mutable)
- identified by QName
  - namespace of the containing module

## Extensions to Prolog - Collections

---

```
module namespace n = "http://news.org/";  
  
declare collection n:articles as node()*;  
  
declare ordered collection n:months  
    with read-only nodes;
```

## Function Library - Collections

---

```
import module namespace n = "http://news.org/";  
  
xqddf:create-collection(xs:QName("n:articles"))  
  
xqddf:collection(xs:QName("n:articles"))  
  
xqddf:insert-nodes-last(  
    xs:QName("n:articles"),  
    <article>...</article>)  
  
xqddf:is-collection-declared(  
    xs:QName("n:articles"))
```

# Indexes

---

- key - node mappings
  - multiple keys
  - node needs to be stored in a collection
- properties
  - type (e.g. value equality/range)
  - unique key
  - maintenance (automatic vs. manual)
- identified by QName

## Extensions to Prolog - Indexes

---

```
module namespace n = "http://news.org/";  
  
(: employees collection decl :)  
  
declare value equality index n:CityEmp  
  on nodes  
  xqddf:collection(xs:QName("n:employees"))  
by  
  .//news:station/news:city as xs:string;
```

## Function Library - Indexes

---

```
import module namespace n = "http://news.org/";  
  
xqddf:create-index(xs:QName("n:CityEmp"))  
  
xqddf:probe-index-point(  
    xs:QName("n:CityEmp"), "Prague"  
)  
  
xqddf:refresh-index(xs:QName("n:CityEmp"))  
  
xqddf:is-index-available(xs:QName("n:CityEmp"))
```

# Integrity Constraints (ICs)

---

- make sure that data in collection is
  - accurate
  - consistent
- types of ICs
  - for each / all nodes
  - foreign key
- checked at applyUpdates (after validation)



## Extensions to Prolog - Integrity Constraints (I)

---

```
module namespace n = "http://news.org/";  
  
(: employees collection decl :)  
  
declare integrity constraint n:UniqueId  
  on collection n:employees  
  node $emp check unique key $emp/@id;
```

## Extensions to Prolog - Integrity Constraints (2)

---

```
module namespace n = "http://news.org/";

(: articles collection decl :)

declare integrity constraint n:AuthorNames
  on collection n:articles
  foreach node $a
  check string-length($a/author/name) != 0;
```

## Extensions to Prolog - Integrity Constraints (3)

---

```
module namespace n = "http://news.org/";  
  
(: articles & employees collection decl :)  
declare integrity constraint n:AuthorExists  
foreign key  
from collection  
    n:articles node $x key $x/empid  
to collection  
    n:employees node $y key $y/@id;
```

## Conclusion & Outlook

---

- XQuery Data Definition Facility
  - collections, indexes, and integrity constraints
  - processing model, prolog, function library
- Implemented in Sausalito 1.0 & Zorba 1.0
  - XML end-to-end architectures
- Consider taking into core language
  - 90% standard is no standard
- Reviewers & More implementations

Thank you!

---

<Thanks/>

Comments? Suggestions?

[matthias.brantner@28msec.com](mailto:matthias.brantner@28msec.com)



**ETH**

Eidgenössische Technische Hochschule Zürich  
Swiss Federal Institute of Technology Zurich

**28msec**

**ORACLE®**

---