

# 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

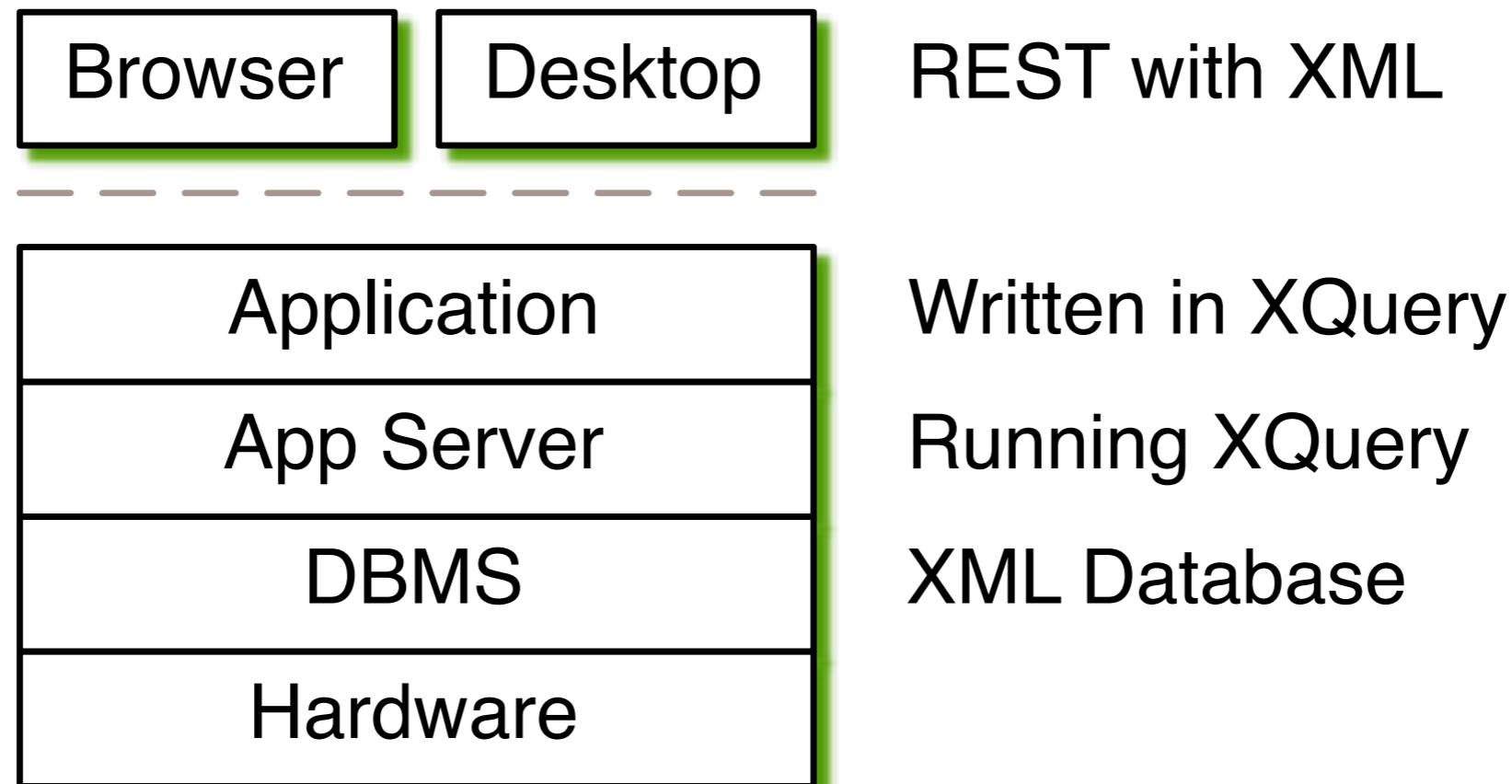
**28msec**

**ORACLE®**

## Context

---

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



## Motivation & Goal

---

- XQuery is incomplete
  - model, describe, and reason about the database
  - semantics of collections (e.g. copy or order)
  - statically declare and manage
    - collections, indexes, integrity constraints (ICs)
- Extension to XQuery 1.1 and XQuery Update
- XQuery Data Definition Facility
-

## Goal

---

- Extension to XQuery 1.1 and XQuery Update
  - XQuery Data Definition Facility (XQDDF)
- Collections, indexes, and integrity constraints
  - declaration
  - lifetime
  - usage
  - sharing by multiple programs

# Approach

---

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

# Extensions to the Processing Model

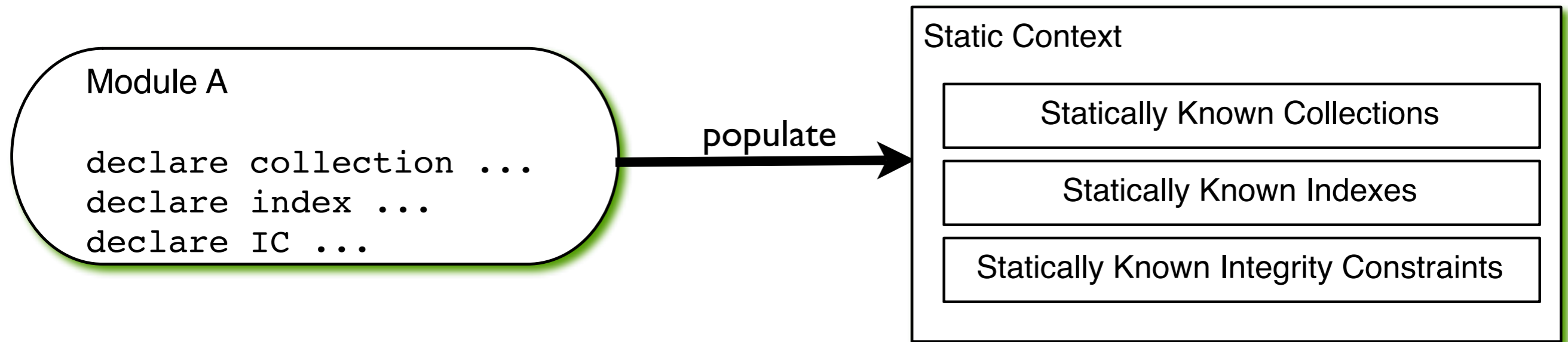
---

Module A

```
declare collection ...  
declare index ...  
declare IC ...
```

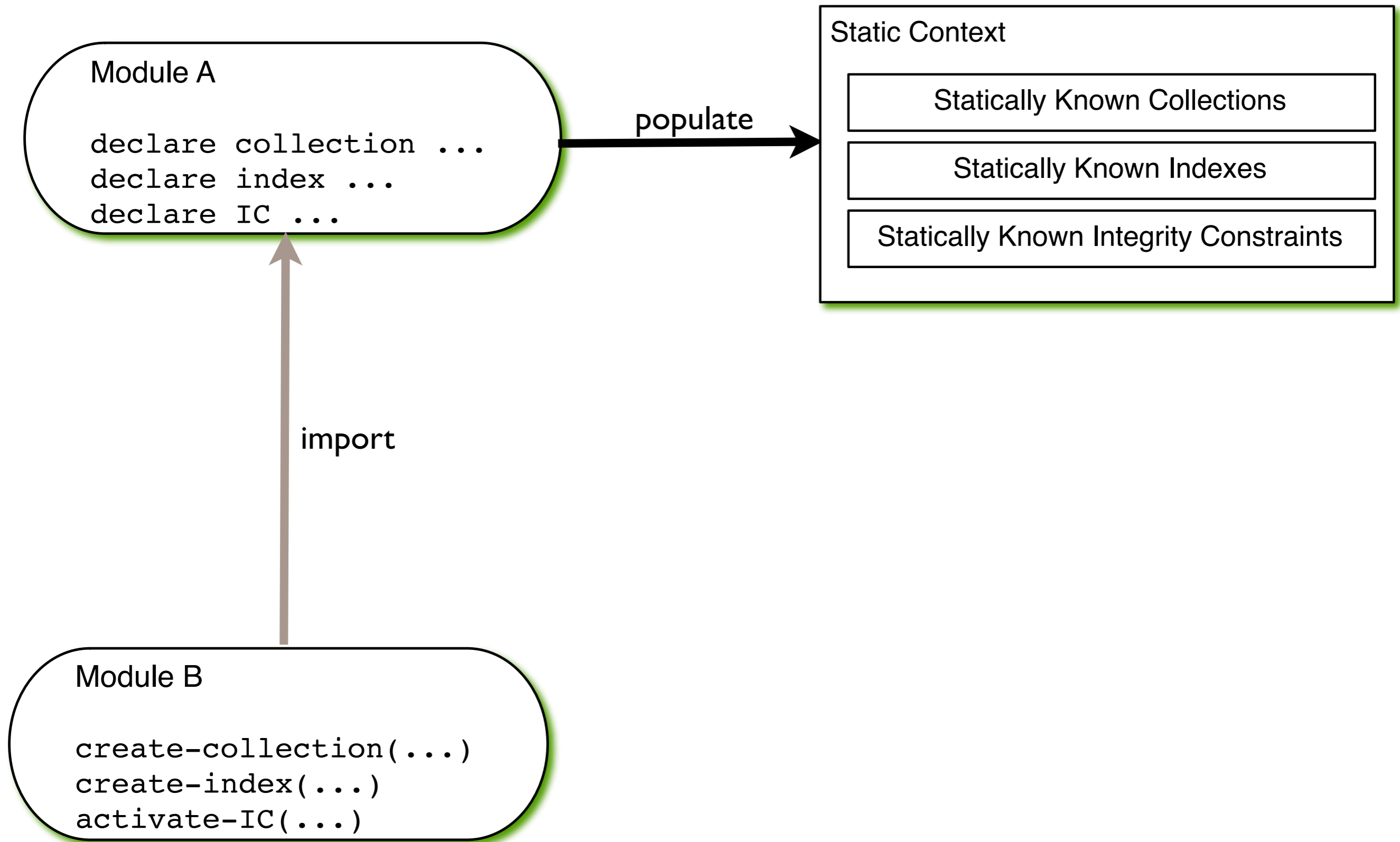
# Extensions to the Processing Model

---



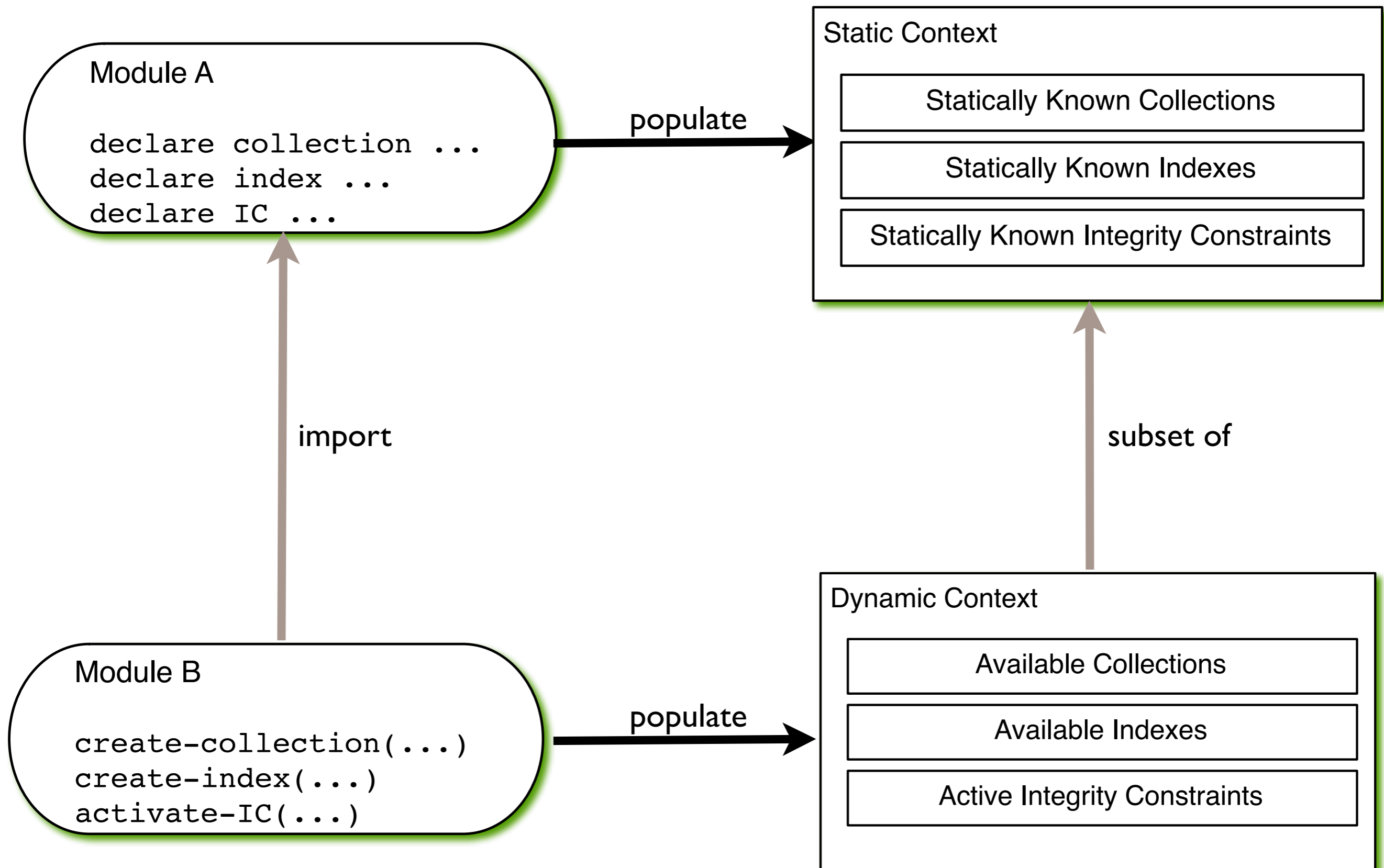
# Extensions to the Processing Model

---



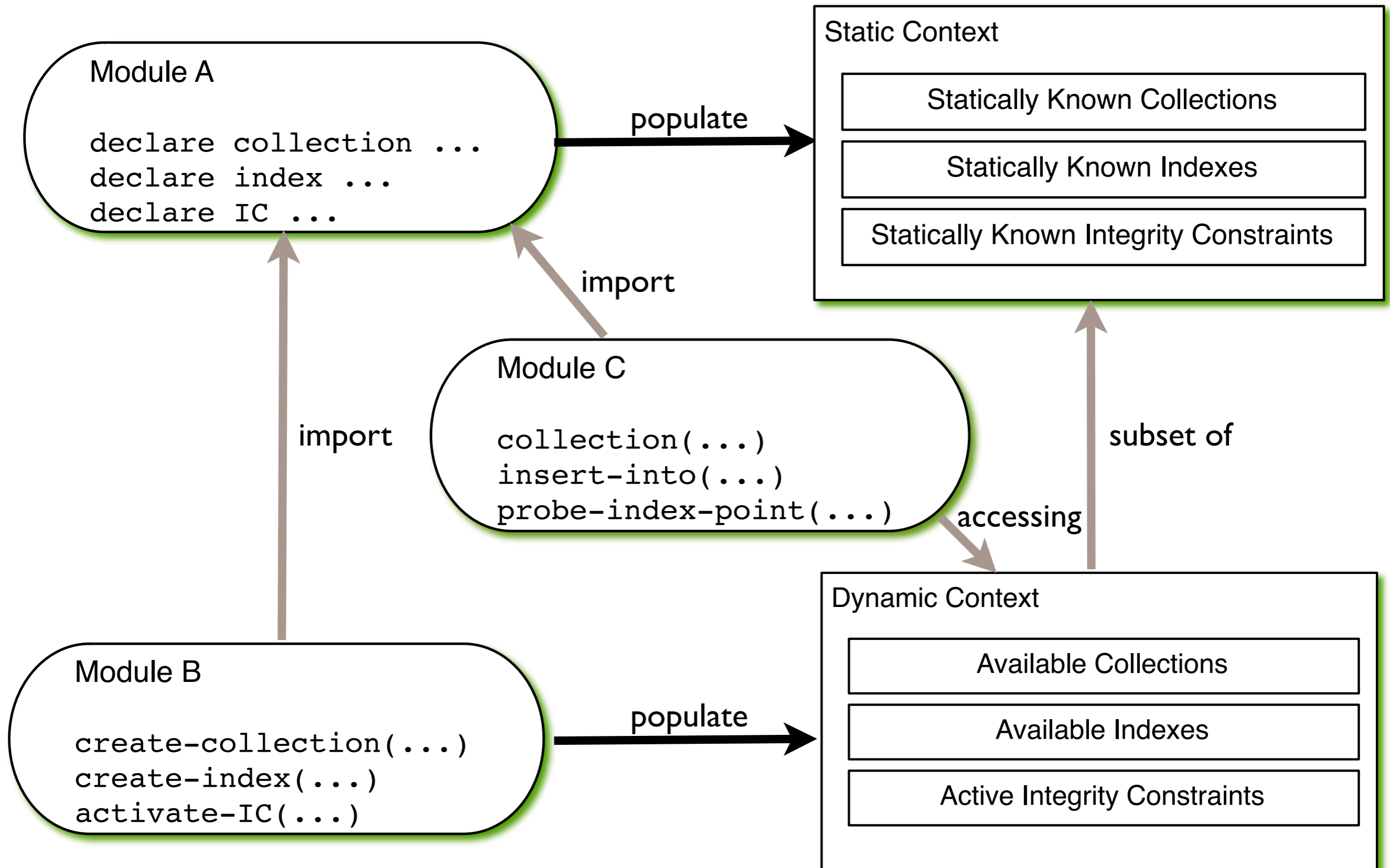
# Extensions to the Processing Model

---



# Extensions to the Processing Model

---



# Collections

---

- sequences of parent-less nodes
- disjoint
- properties
  - types
  - properties
- identified by QName

## 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:insert-nodes-last(  
  xs:QName("n:articles"),  
  <article>...</article>  
)
```

```
xqddf:collection(xs:QName("n:articles"))
```

# Indexes

---

- key - node mappings
  - multiple keys
  - nodes need to be stored in a collection
- properties
  - equality vs. range
  - maintenance (automatic vs. manual)

## Extensions to Prolog - Indexes

---

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

(: 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"))
```

# Integrity Constraints (ICs)

---

- make sure that data in collection is
  - accurate
  - consistent
- types of ICs
  - entity (all nodes)
  - domain (each node)
  - referential (foreign key)
- checked at applyUpdates

## Extensions to Prolog - Integrity Constraints (I)

---

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

```
(: collection decl :)
```

```
(: entity ic :)
```

```
declare integrity constraint n:UniqueId
```

```
on collection n:employees
```

```
node $emp check unique key $emp/@id;
```

## Extensions to Prolog - Integrity Constraints (I)

---

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

```
(: collection decl :)
```

```
(: entity ic :)
```

```
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/";

(: collection decl :)

(: domain ic :)

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/";

(: referential ic :)

declare integrity constraint n:AuthorNames

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
- Take into core language
  - 90% standard is no standard
- More implementations

Thank you!

---

<Thanks/>

Comments? Suggestions?

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



**ETH**

Eidgenössische Technische Hochschule Zürich  
Swiss Federal Institute of Technology Zurich

**28msec**

**ORACLE®**

---