



# Entwurf Verteilter Systeme

**Prof. Dr.-Ing. Martin Gaedke**

Technische Universität Chemnitz

Fakultät für Informatik

Professur Verteilte und selbstorganisierende  
Rechnersysteme

<http://vsr.informatik.tu-chemnitz.de>



## Part III

# THE WEB AS A PLATFORM FOR DISTRIBUTED APPLICATIONS



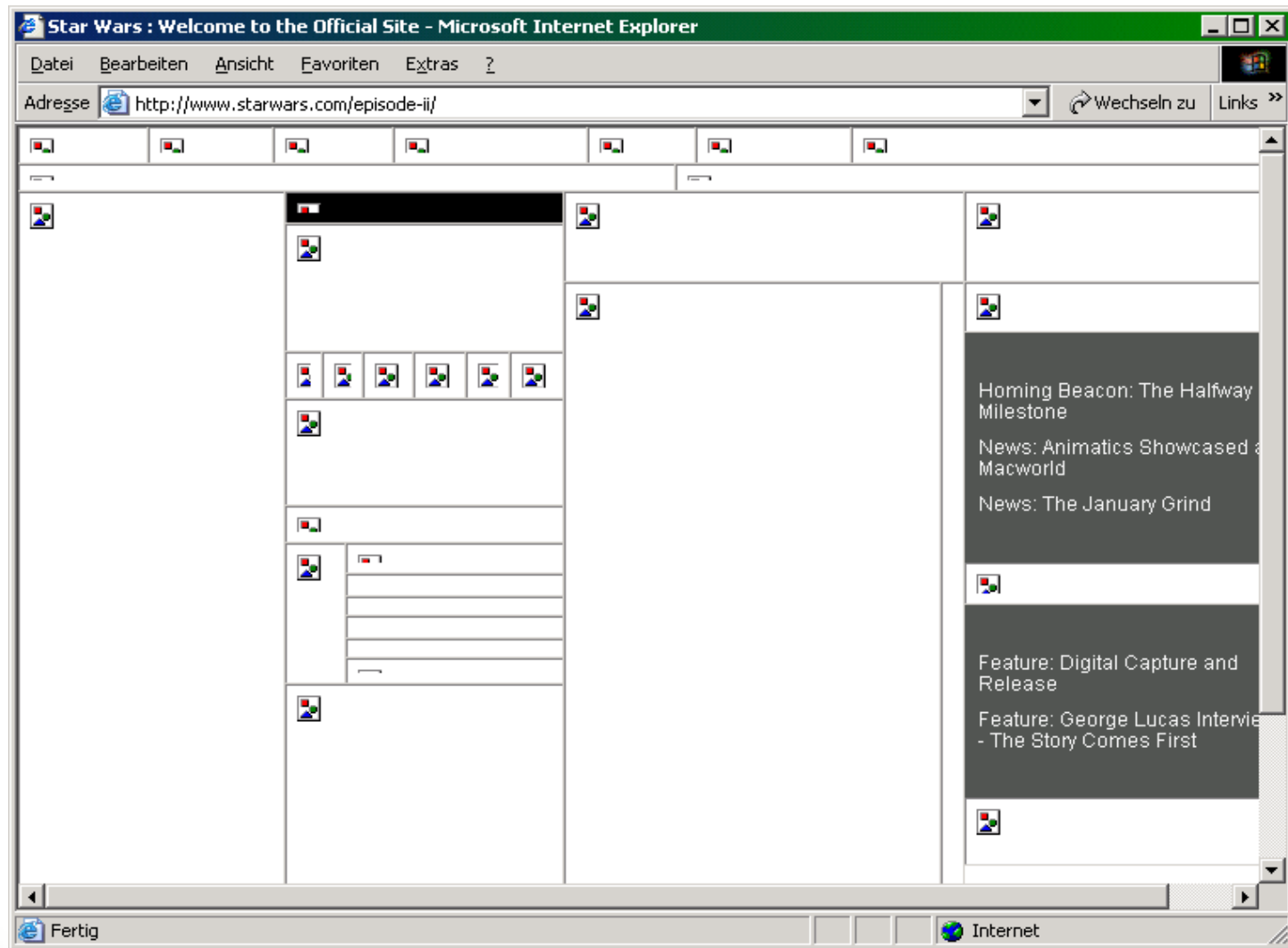
# Section

# **WEB SYSTEM BASICS**

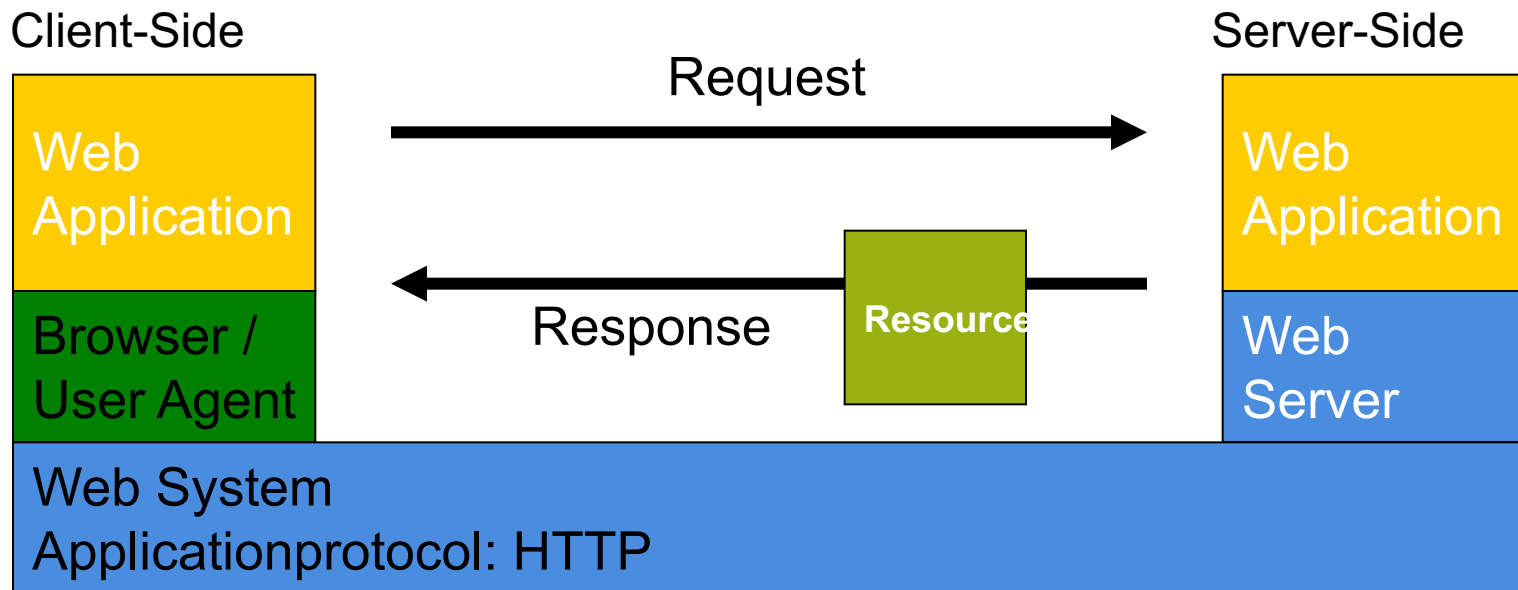


# Return Of The Resource (1)

Screenshot from the site [www.starwars.com](http://www.starwars.com)



# 1<sup>st</sup> Generation of the World-Wide-Web



## ■ Browser

- ▶ Mosaic
- ▶ HTML
- ▶ Images (GIF)
- ▶ HTML-Forms
- ▶ Helper
  - Audio, Video etc.

## ■ Web System

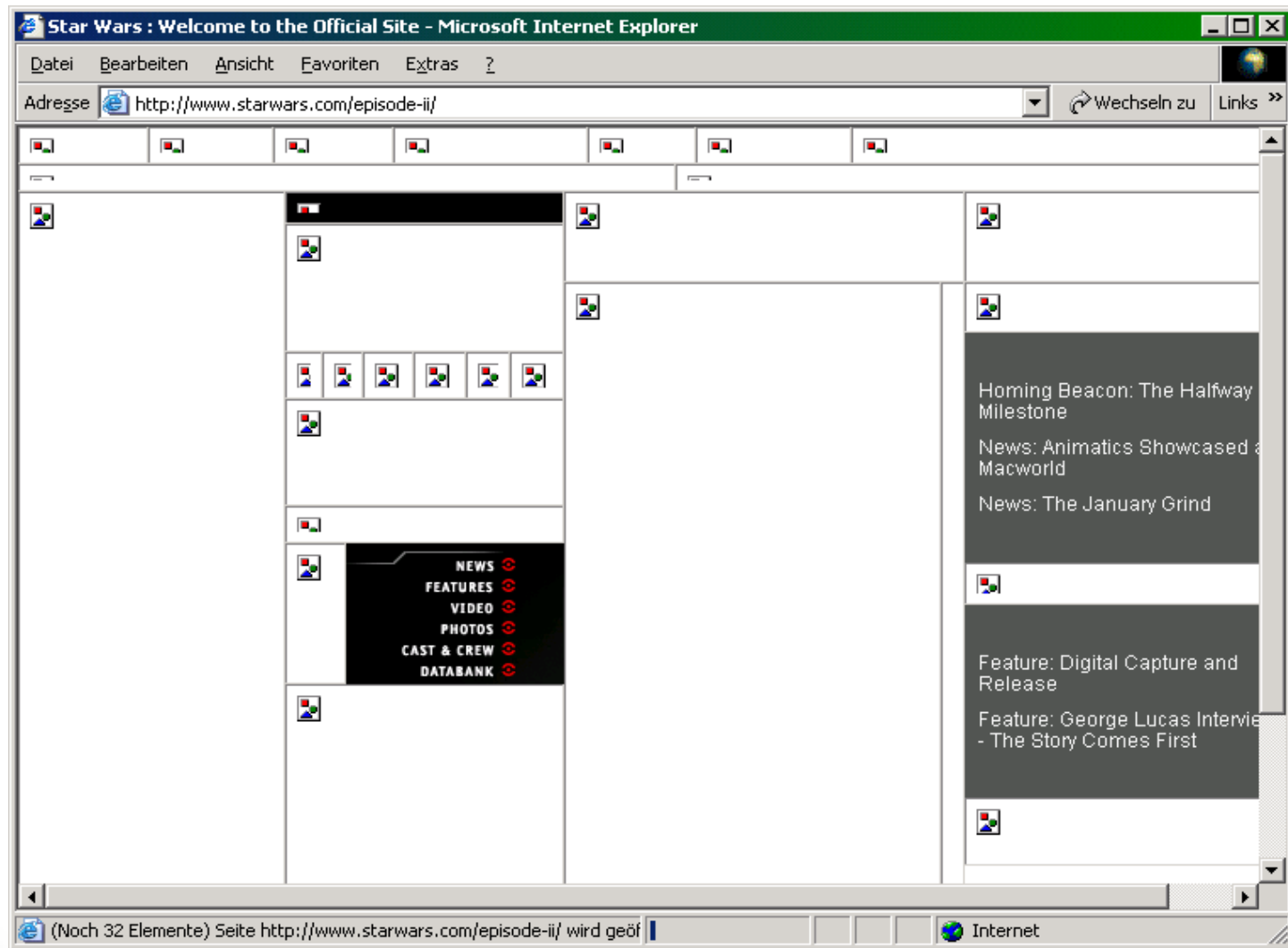
- ▶ HTTP

## ■ Web Server

- ▶ HTTP
- ▶ CGI
  - Database
  - Information Systems

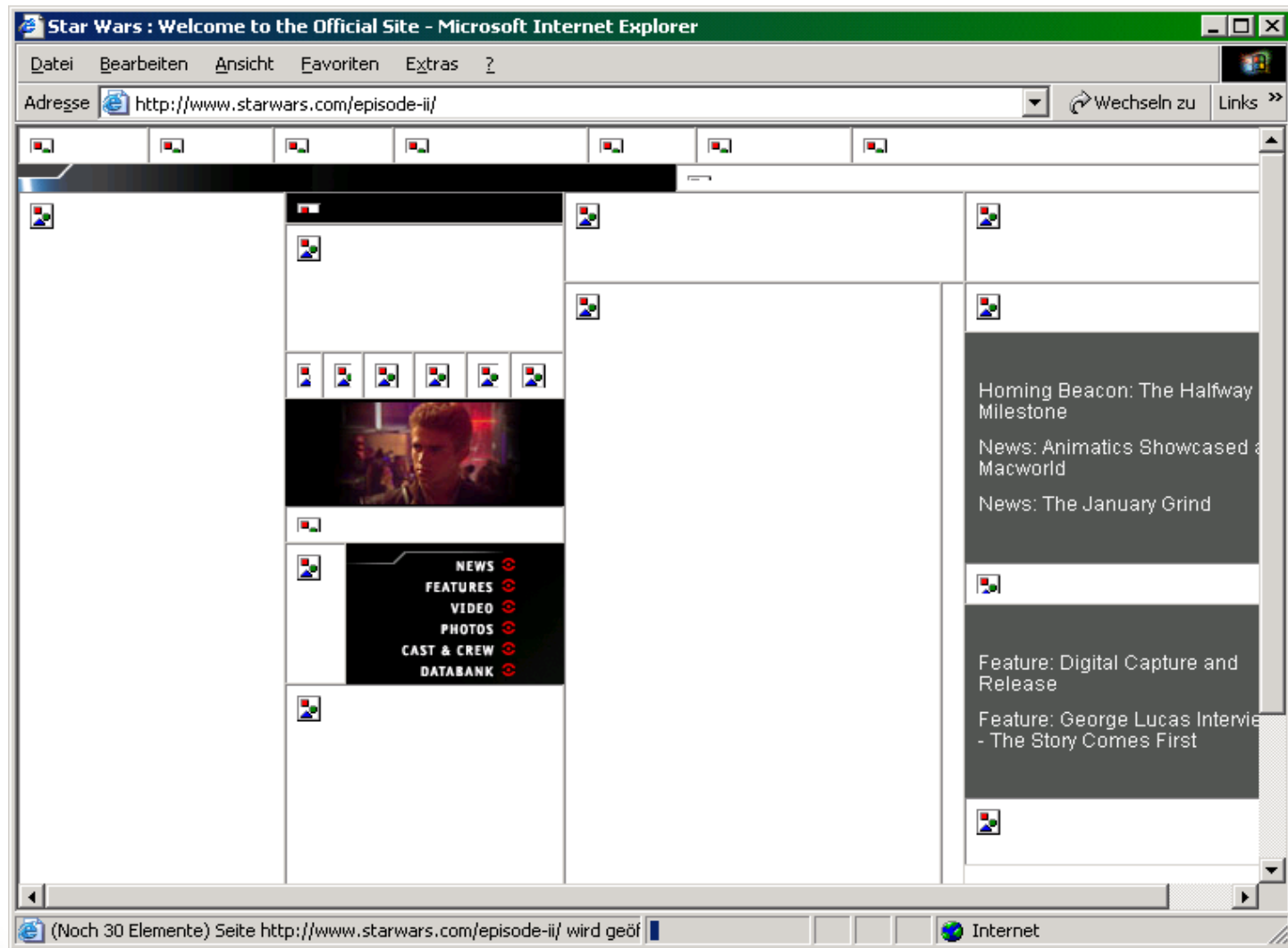
# Return Of The Resource (2)

Screenshot from the site [www.starwars.com](http://www.starwars.com)



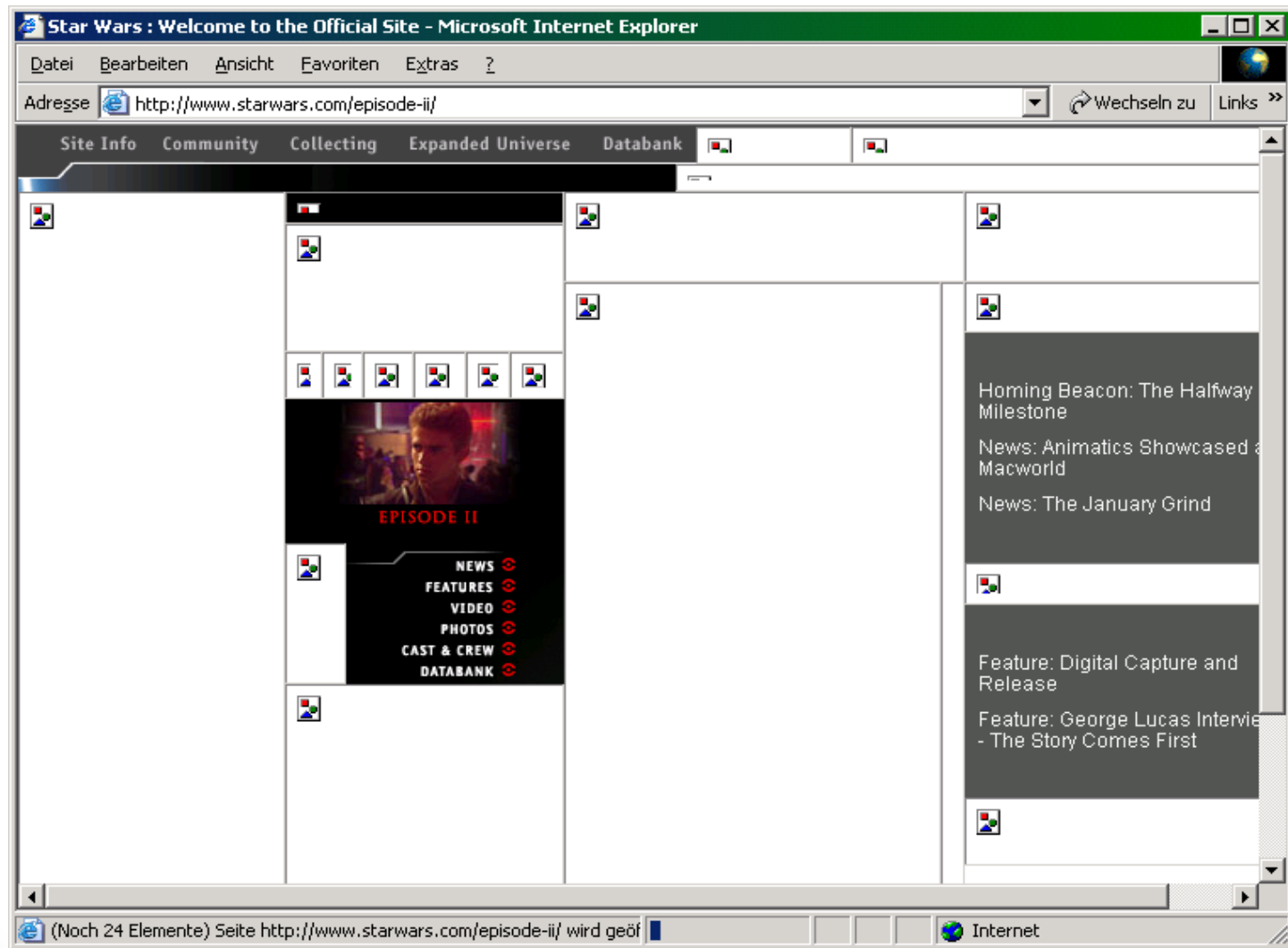
# Return Of The Resource (3)

Screenshot from the site [www.starwars.com](http://www.starwars.com)



# Return Of The Resource (4)

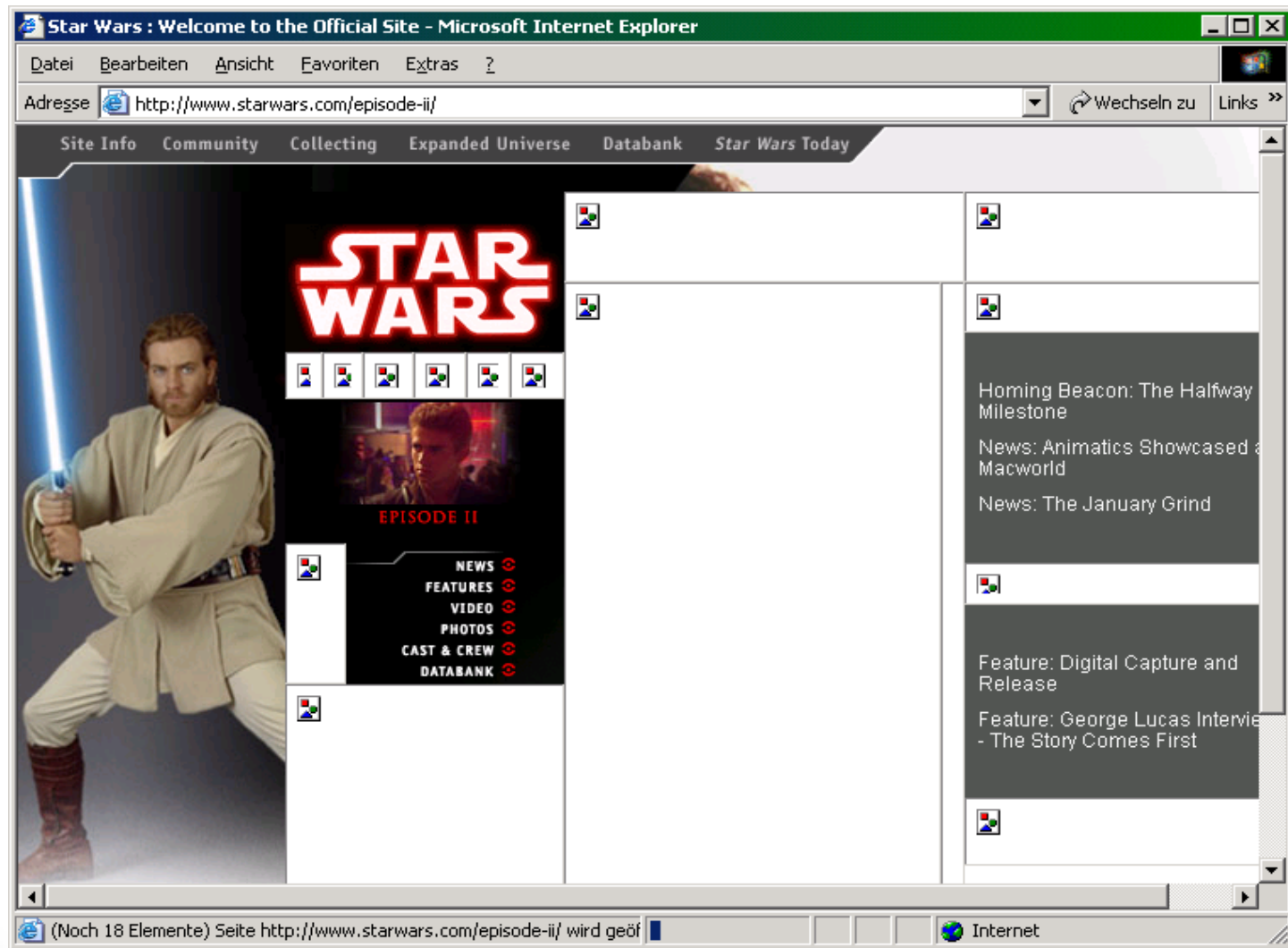
Screenshot from the site [www.starwars.com](http://www.starwars.com)





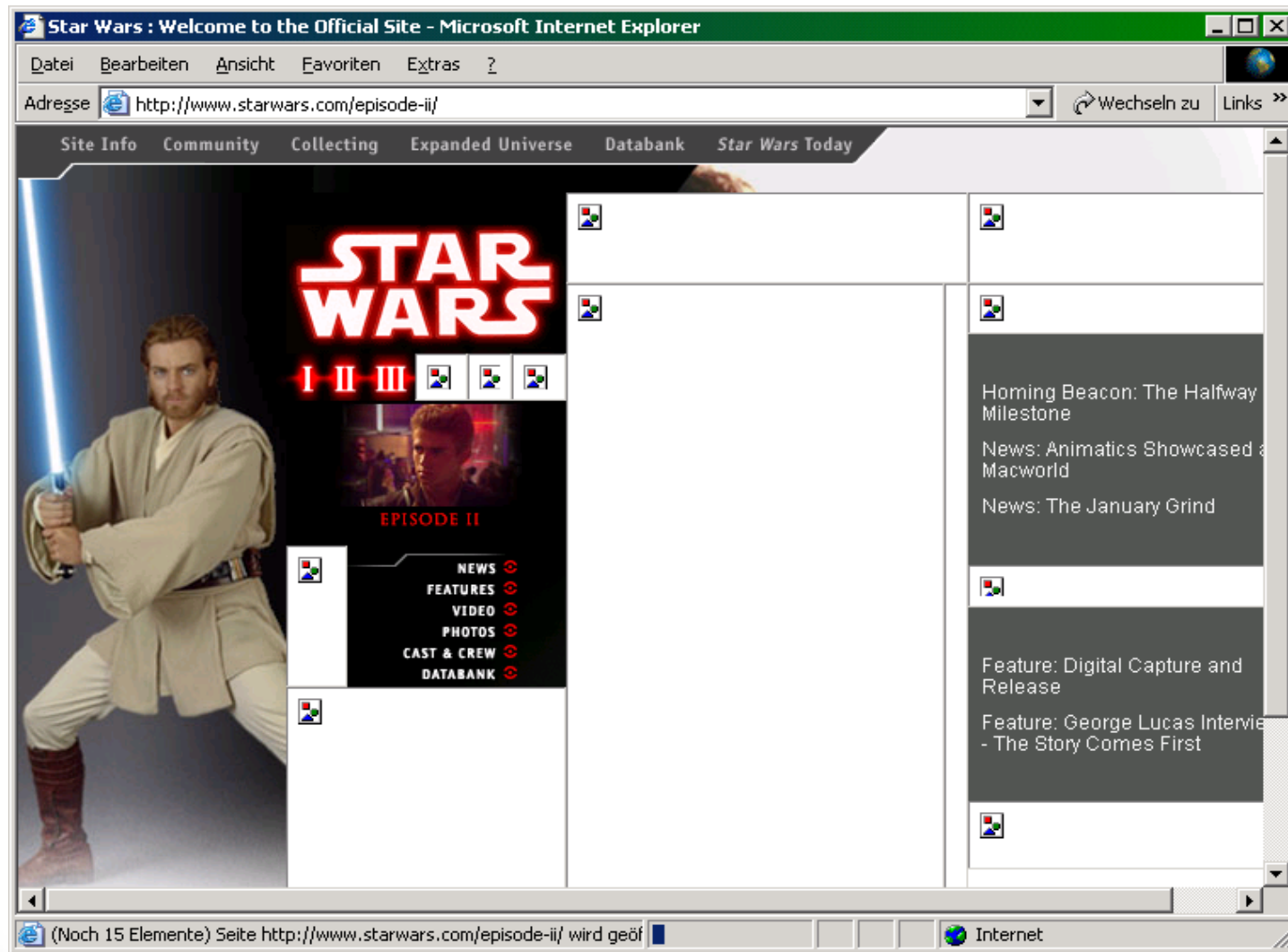
# Return Of The Resource (5)

Screenshot from the site [www.starwars.com](http://www.starwars.com)



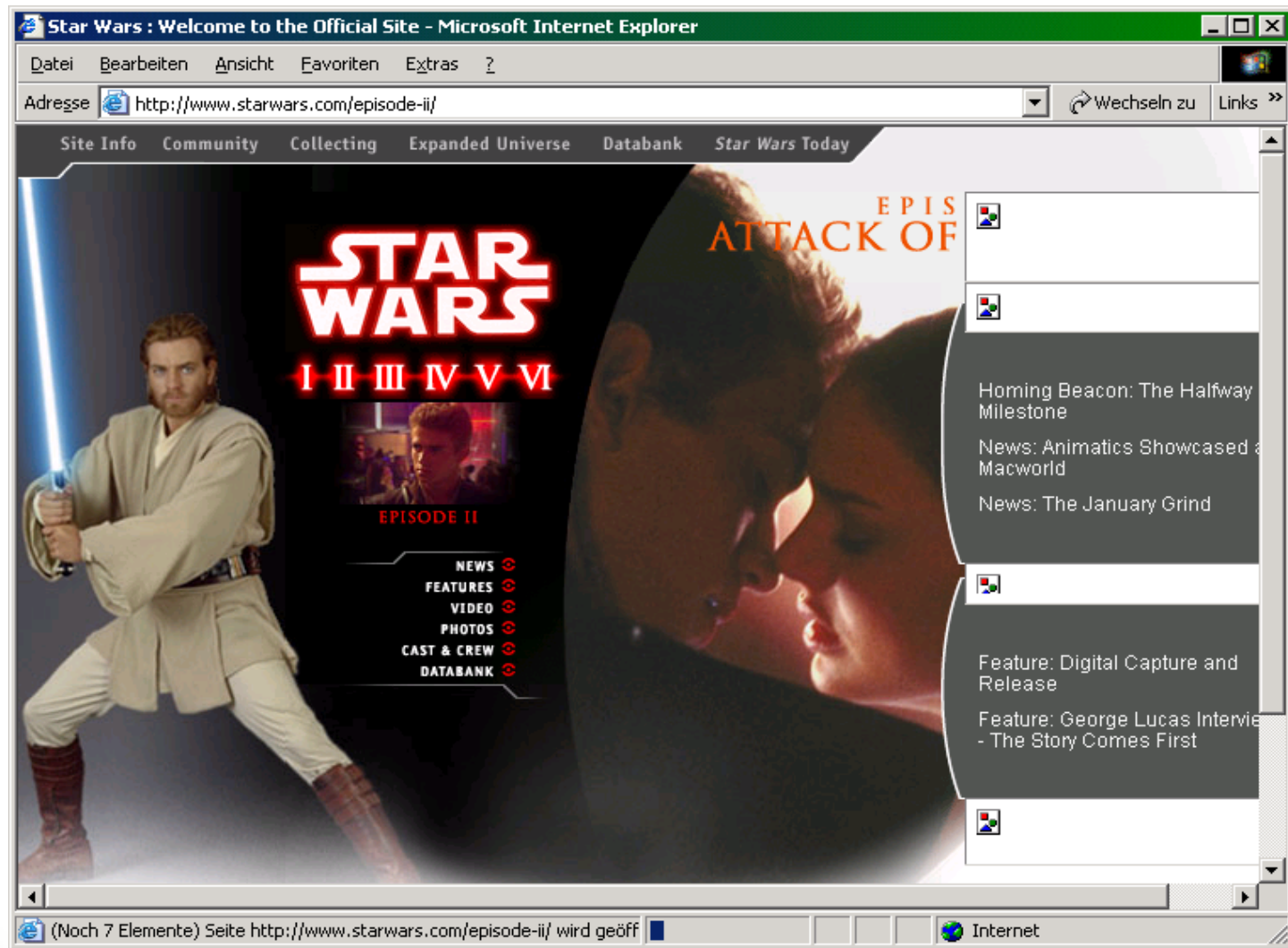
# Return Of The Resource (6)

Screenshot from the site [www.starwars.com](http://www.starwars.com)



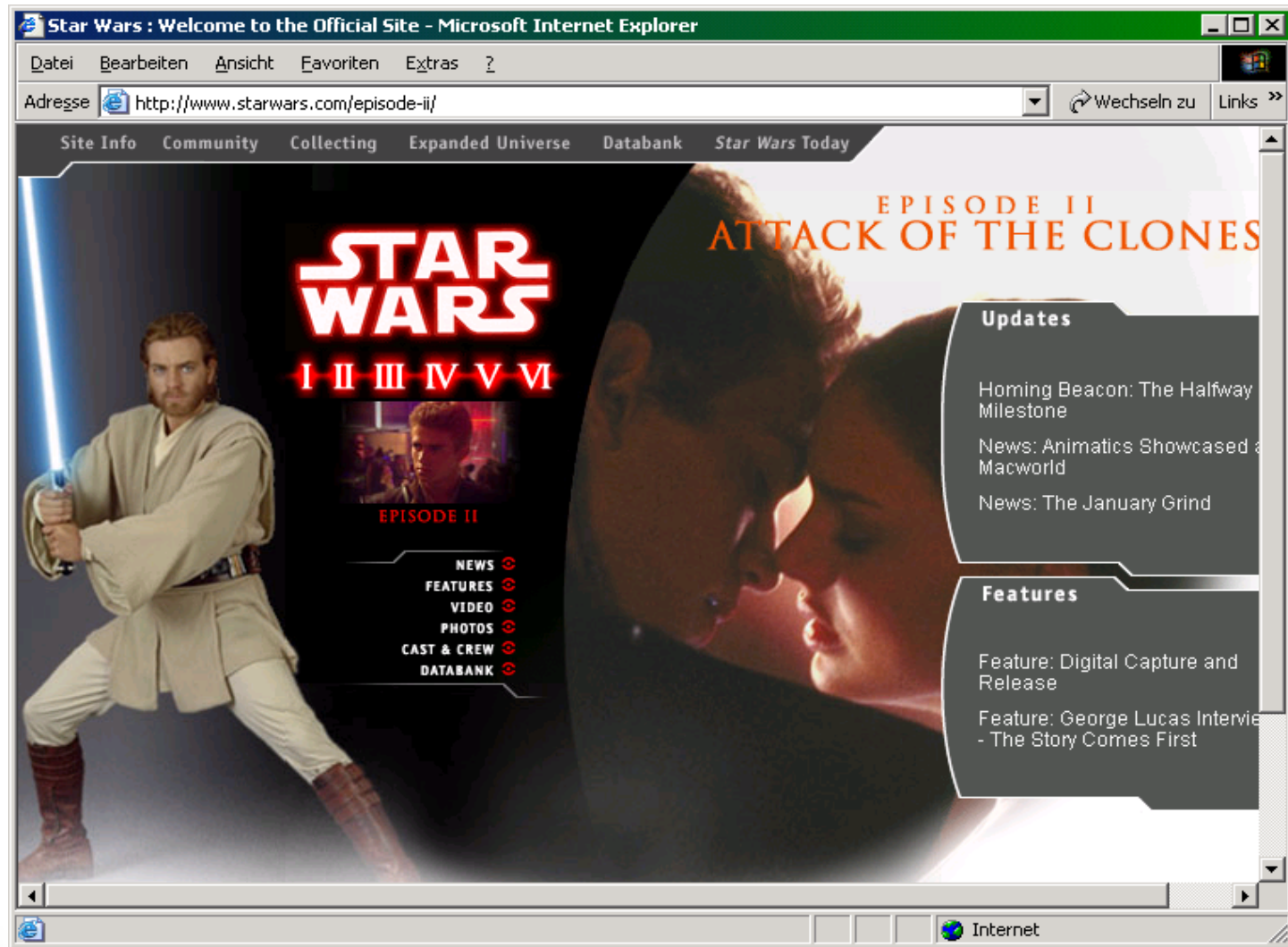
# Return Of The Resource (7)

Screenshot from the site [www.starwars.com](http://www.starwars.com)



# Return Of The Resource (8)

Screenshot from the site [www.starwars.com](http://www.starwars.com)



# Retrieving Information

## Client-Side:

- (1) Prepare Request
- (2) Request Resource
- (3) Handle Response
- (4) Process/Render  
Data of Resource

## Server-Side:

- (A) Handle Request
- (B) Process
- (C) Send Resource



# 1 - Prepare Request

---

- Address the resource (URI)
  - Example: **http://localhost/**
- Find URI-Resolver for scheme in use
  - Example: URI-Resolver for **http**
- URI Resolver
  - Get address of resource (scheme-specific)
  - Example **URL-Resolver:**
    - Host: localhost
    - Resource: /



# 2 - Request Resource

---

- Send request to address (**Communication**)

- Depends on scheme of URI, e.g. mailto, http
- Transmission Protocol defined by scheme
- Remember: allows for caching!
- Example:

Use TCP, connect with localhost at port 80 and send:

GET / HTTP/1.0

[CRLF]





# Request Resource – Demo

- URI: http://localhost/
- **URL-Resolver:**
  - Scheme=http
  - Transmission-Protocol: TCP/IP
- Scheme-specific-part = localhost
  - Check scheme-specific-part for Host, retrieve IP using DNS
  - IP Address for localhost: 127.0.0.1
  - Port: 80 (if not specified by scheme-specific-part)
- Send Protocol Code

```
Telnet> set LOCAL_ECHO
Telnet> open 127.0.0.1 80
```

```
GET / HTTP/1.0
```

```
-
```



# 3 - Handle Response

- Handle protocol, e.g. HTTP 302 Object moved, cookies
- Retrieve resource
- Example:
  - Further action needed
  - Response includes resource



```
HTTP/1.1 302 Object moved
Server: Microsoft-IIS/5.0
Date: Wed, 23 Jan 2002 23:50:11 GMT
Location: localstart.asp
Connection: Keep-Alive
Content-Length: 161
Content-Type: text/html
Set-Cookie: ASPSESSIONIDGGGQQUFU=HENLKFKDIEMP
Cache-control: private

<head><title>Objekt verschoben</title></head>
<body>Bitte klicken Sie hierauf, um das Objekt <a HREF="localstart.asp">anzusehen</a></body>
```

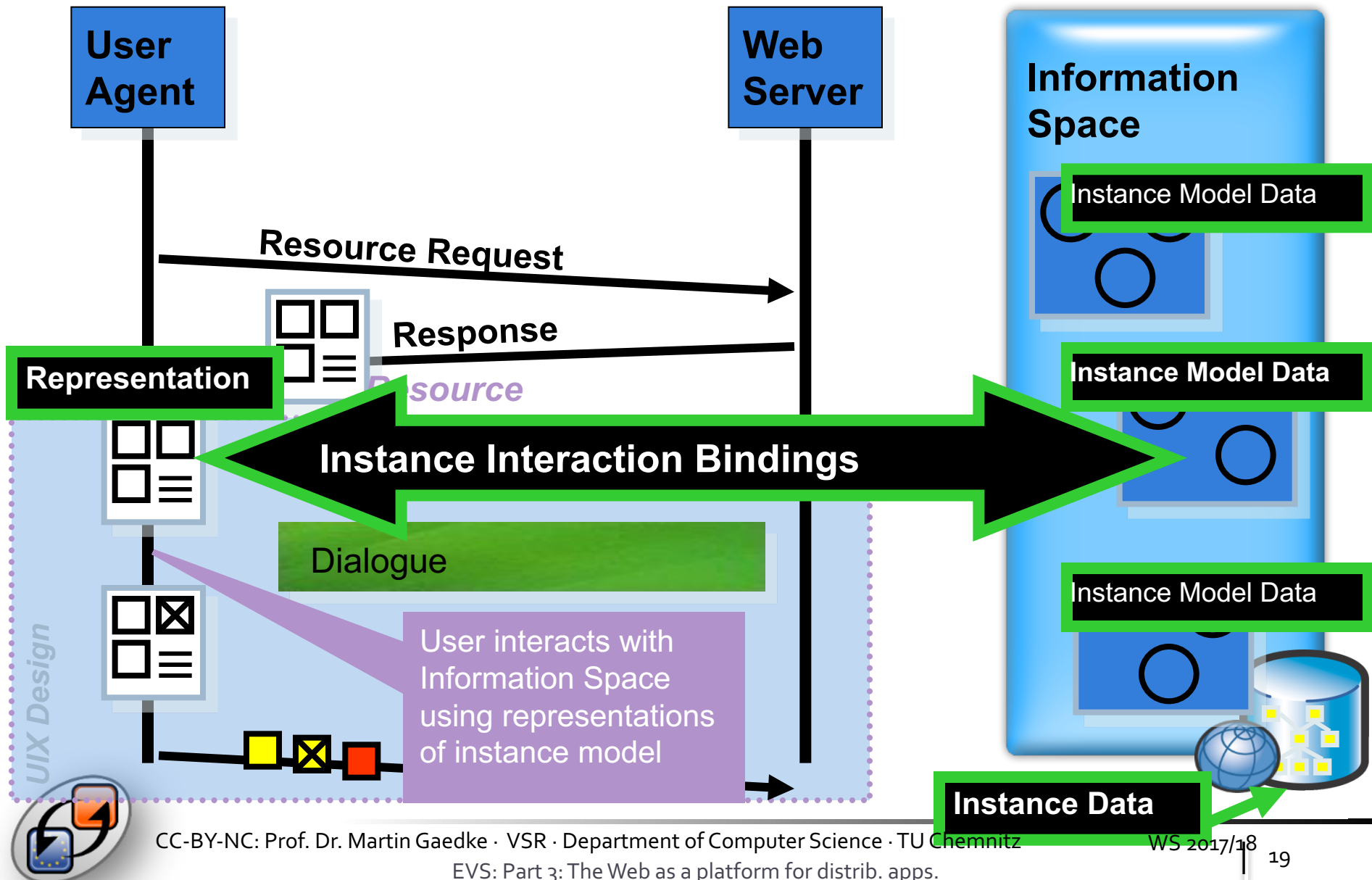
# 4 - Process Data of Resource

---

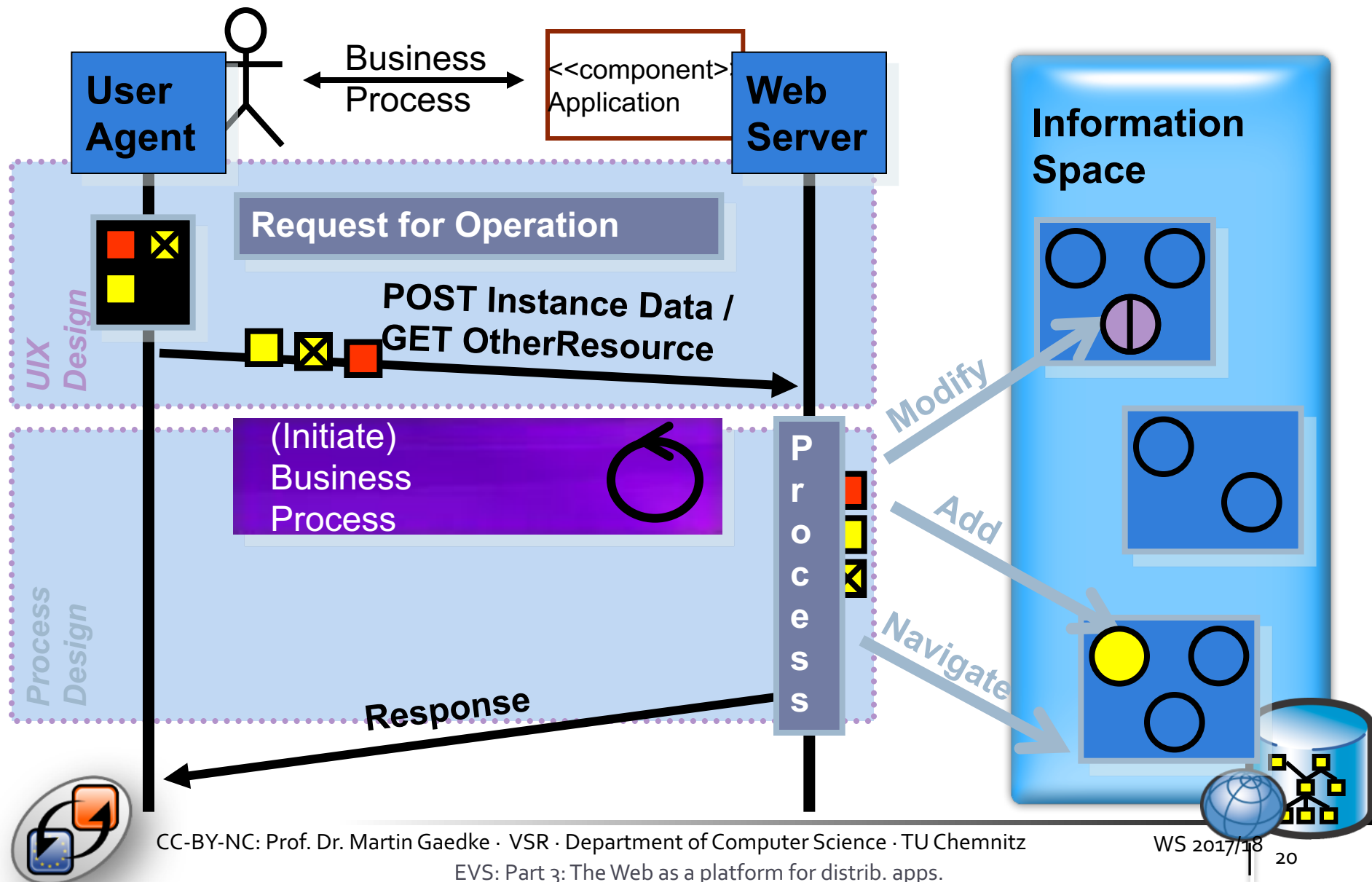
- Processing depends on user agent
- Browser
  - Process header
  - Check content-type
  - Process resource data depending on MIME-Type
  - E.g. render: text/html, text/text, image/gif
- Other User Agents
  - Process header
  - Process data
  - E.g. WebService-Client: Process XML-Resource, wget/robots retrieving collections of Web-Pages



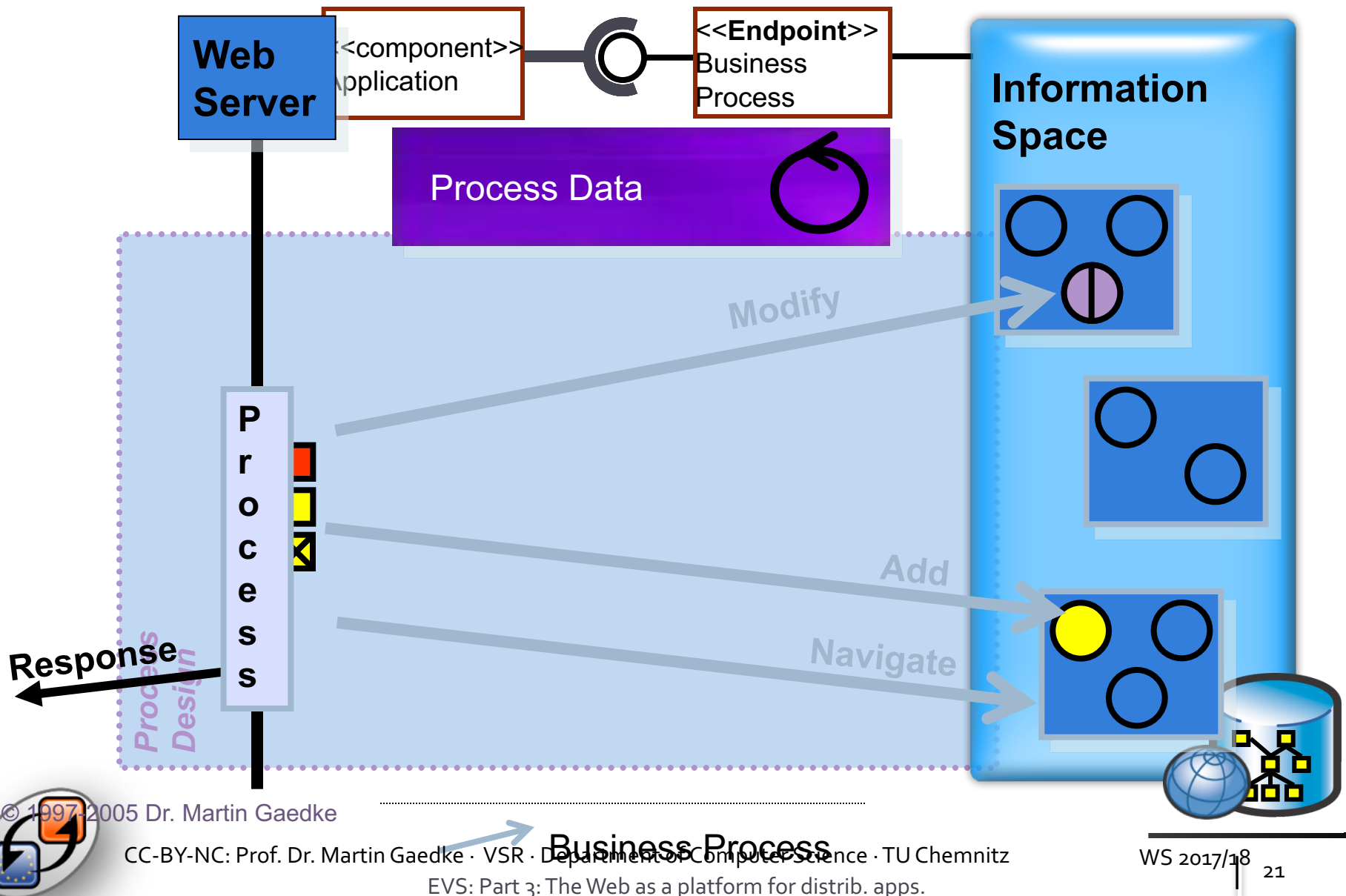
# Interaction Model in the Web



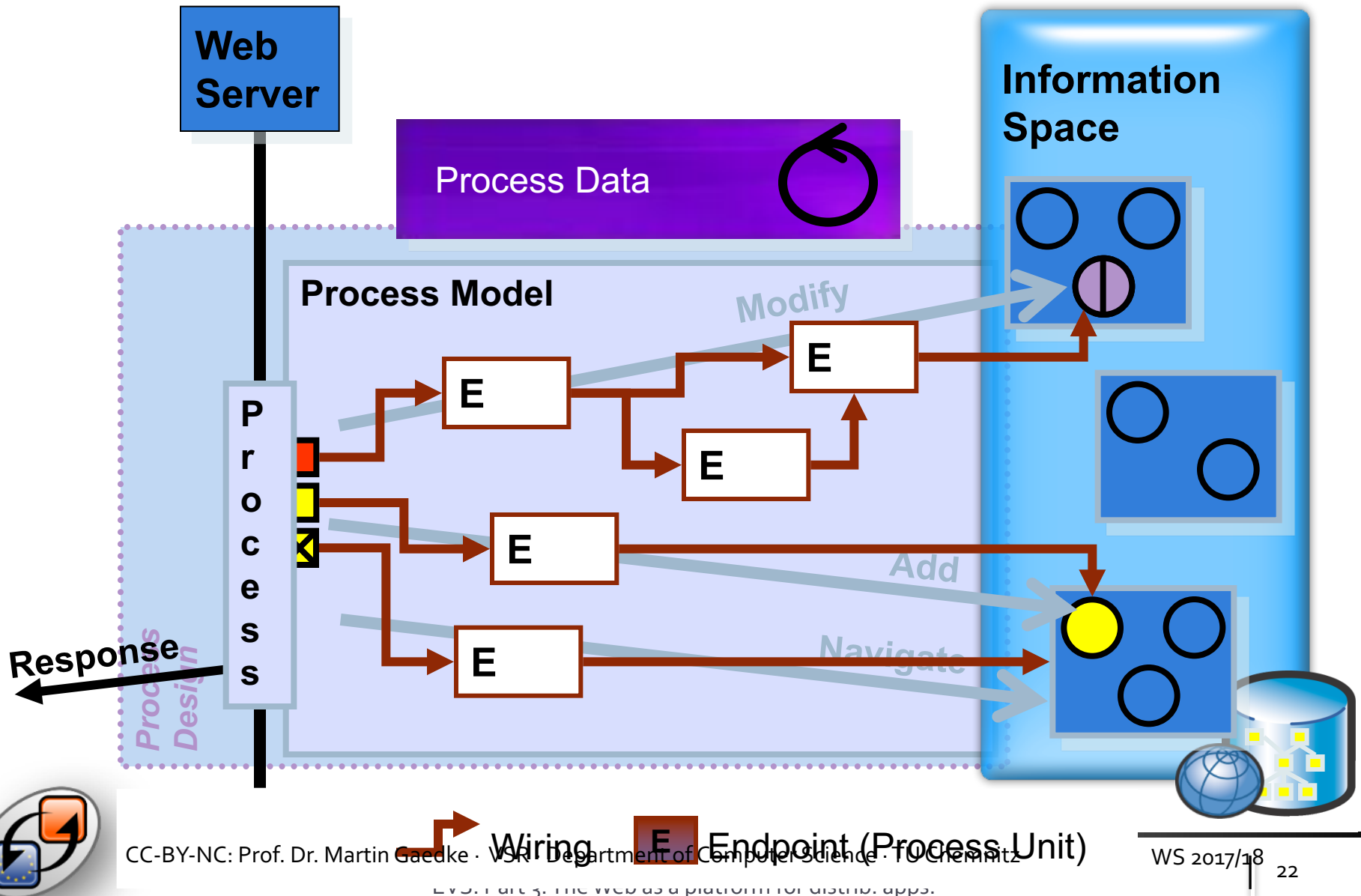
# Process and the Web



# Process Model and the Web



# Process Model in Detail



# Section

# UNIFORM ADDRESSING



# Addressing Resources

---

- As with DNS and IP-Adresses...
- Goals:
  - It must be possible to identify resources
    - By Name
    - By Address resp. Location
  - Any resource in the Internet should be identified
    - Web pages, FTP-Resources, Mailboxes, Directories, interactive services
- Requirements: Identification mechanism should be
  - Extensible
  - Complete
  - Printable (to be represented as string of 7-bit characters)





# Uniform Resource Identifier

---

- Uniform Resource Identifier (URI)
  - Generic term for all textual names/addresses
  - URI is URL or URN or URC
- Uniform Resource Locator (URL)
  - The set of URI schemes that have explicit instructions on how to access the resource over the Internet
- Uniform Resource Name (URN)
  - A URI that has an institutional commitment to availability, etc.
  - A particular scheme intended to identify resources
- Uniform Resource Characteristic (URC)
  - A URC provides Meta Information



# Uniform Resource Identifier

---

- **URI – Syntax for identifiers [RFC3986]**

$\langle \text{URI} \rangle ::= \langle \text{scheme} \rangle " : " \langle \text{scheme-specific-part} \rangle$

- **$\langle \text{scheme} \rangle$**

name of the scheme

- **$\langle \text{scheme-specific-part} \rangle$**

identifier in a format that is according to the scheme



# Reserved Characters

---

- For all types of URIs the following Rules apply:
- The percent sign ("% ", ASCII 25 hex)
  - Escape character
- Hierarchical forms ("/", ASCII 2F hex)
  - Delimiting of substrings whose relationship is hierarchical
- Hash - fragment delimiter ("#", ASCII 23 hex)
  - Identifies a fragment in a resource
- Query Delimiter ("?", ASCII 3F hex)
  - To delimit the boundary between the URI of a query able object



# Uniform Resource Name

---

- URN – Scheme definition [RFC 1737, RFC 2141]
  - URNs serve as persistent, location-independent, resource identifiers
- `<scheme> ::= "urn"`
- `<scheme-specific-part> ::= <nid> ":" <nss>`
  - nid = Namespace Identifier
  - nss = Namespace Specific String
- E.g. `urn:schemas:httpmail:subject`



# URN Properties

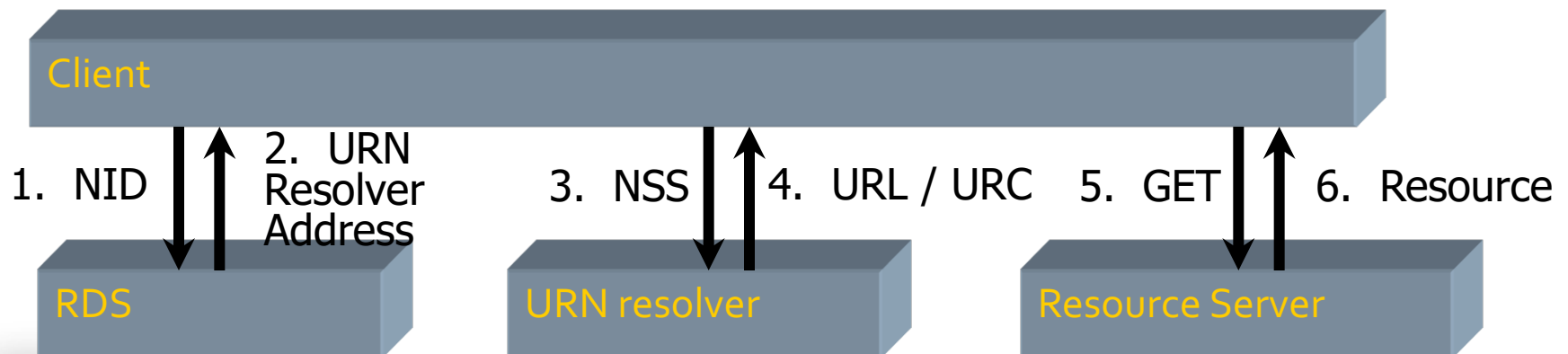
---

- Global scope and uniqueness
- Persistence
- Scalable
- Legacy support
- Extensible
- Independent
- Resolvable



# URN - Resolution

- Infrastructure for URNs is still experimental
  - Resolver Discovery Service (RDS)
  - Name service, name resolution (URN resolver)
  - Result of the resolution is a URL or a URC
  - Cf. RFC 1737, 2276
  - $\langle \text{urn} \rangle ::= \text{"urn:"} \langle \text{nid} \rangle \text{" : " } \langle \text{nss} \rangle$



# Uniform Resource Locator

---

- URL – Scheme definition [RFC1738,3986]
  - *explicit instructions on how to access ...*
  - `<scheme> ::= "http" | "https" | "ftp" | "news" | "mailto" | "nntp" ...`
- Specific Part defined in a general format
  - `<scheme-specific-part> ::=`  
`["//"] [user [":"password] "@"] host [":"port] ["/"url-path]`
- Definitions are maintained by the Internet Assigned Numbers Authority (IANA)
- URLs can also be relative [RFC 1808,3986]



# Example – HTTP URL

- HTTP URL

**<scheme>** ::= "http"

**<scheme-specific-part>** ::= "://"<host>[":"<port>][<abs\_path>]

**<host>** = "[" ( IPv6address | IPvFuture ) "]" | <IPv4address> | <reg-name>

**<port>** = \*DIGIT

**<abs\_path>** = "/"[<path>][";"<params>][?"<query>][#"<fragment>]

**<path>** = <fsegment> \*( "/" <segment> )

- Example

- http://webe.tm.uni-karlsruhe:8080/a/b?x=1#2345
- http://www.secret.xyz/account/euro?add=100#FragId





# Comparison URN vs. URL

	URN	URL
Scope	Global	Global (abs. URL) Local (rel. URL)
Globally Unique	Yes	Yes (abs. URL) No (rel. URL)
Persistent	Yes	No
Scalable	Yes	Yes
Legacy Support	Yes	Limited
Resolution	Not yet determined	Partly using DNS

