

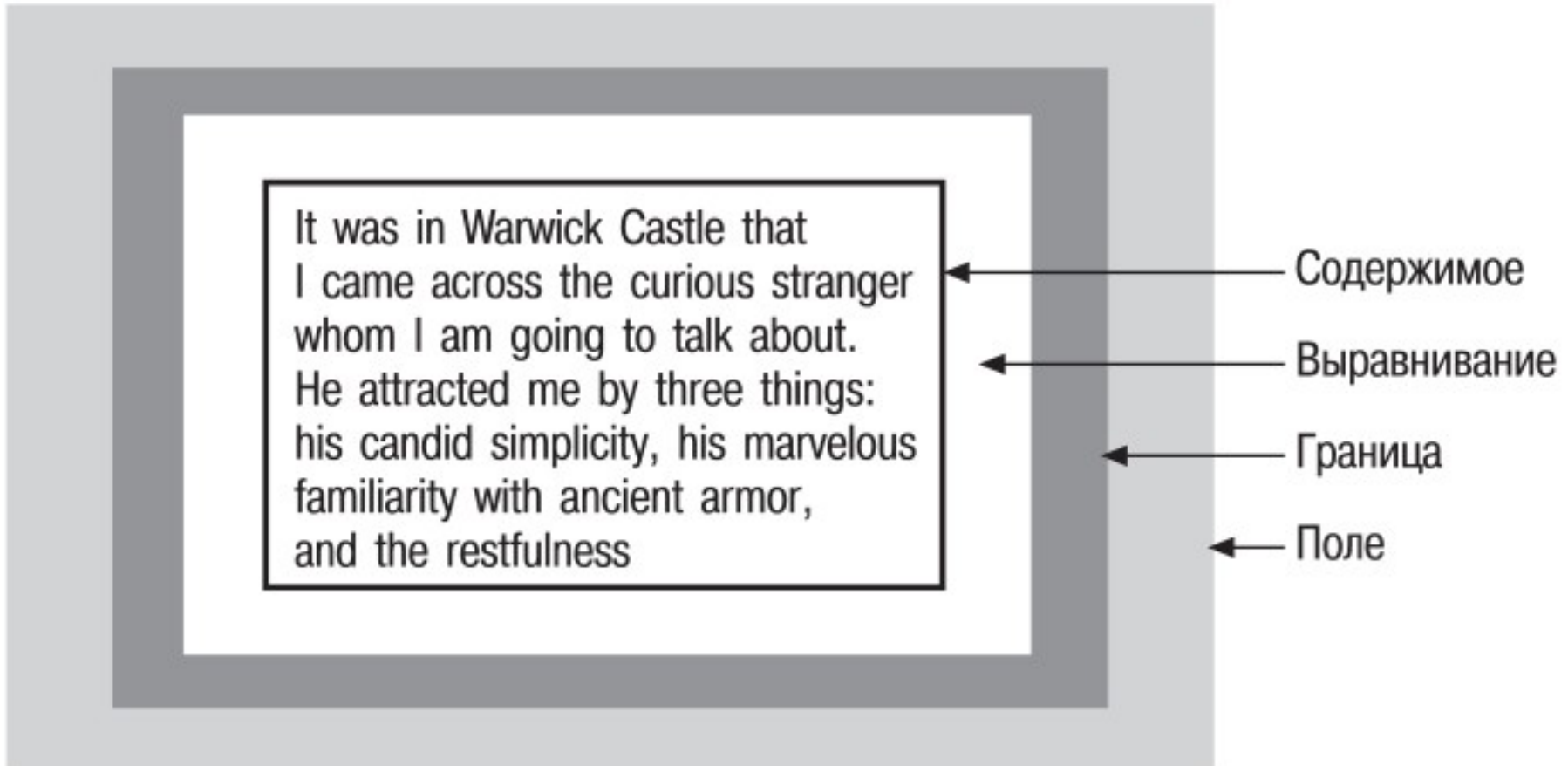
# Web-технологии 3

Форматирующие объекты XSL  
XSL Formatting Objects (XSL-FO)

# Введение

- XSL-FO – это полноценное XML приложение
- Текст + данные в одном документе
- Описание положения прямоугольных блоков на минимум одной странице
- Очень сложное для ручного написания
- XML + XSLT => XSL-FO => HTML, PDF, LaTeX
- Виды блоков:
  - области (regions)
  - блочные зоны (block areas)
  - строковые зоны (line areas)
  - внутри-строчные зоны (inline areas)

# Структура блока



- Размер области XSL-FO равен общему размеру границы, области выравнивания и содержимого
- Оформление текста задается свойствами блоков, включающих этот текст
- Свойства блока определяются атрибутами
- Элементы XSL-FO документа не имеют однозначного соответствия блокам на странице
- Разбиение документа на блоки выполняется в процессе обработки
- Для отображения содержимого генерируется необходимое количество страниц
- XSL-FO документ содержит инструкции и рекомендации

# Структура документа XSL-FO

- Пространство имен  
<http://www.w3.org/1999/XSL/Format>
- Корневой элемент fo:root
- Типичный XSL-FO документ:

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<fo:root
```

```
xmlns:fo="http://www.w3.org/1999/XSL/Format">
```

```
<!-- Элементы форматирующих объектов -->
```

```
</fo:root>
```

- Генерация XSL-FO с помощью XSLT

```
<xsl:template match="/">
```

```
<fo:root>
```

```
<xsl:apply templates/>
```

```
</fo:root>
```

```
</xsl:template>
```

- Обязательные элементы:
  - **fo:layout-master-set** - описание общего формата страницы (макет)
  - **page-sequence** содержит сам текст, помещаемый на страницы, и инструкции по его форматированию

# Макет страницы

- В стандарте определен вид страницы **fo:simple-page-master**
- Прямоугольная страница с фиксированной шириной, высотой и полями со всех четырех сторон
- Имя макета задается атрибутом **master-name**
- В документе может быть несколько макетов

# Пример

```
<fo:simple-page-master master-name="cover"  
  page-height="12cm"  
  page-width="12cm"  
  margin-top="0.5cm"  
  margin-bottom="0.5cm"  
  margin-left="1cm"  
  margin-right="0.5cm">  
</fo:simple-page-master>
```





# Элементы областей

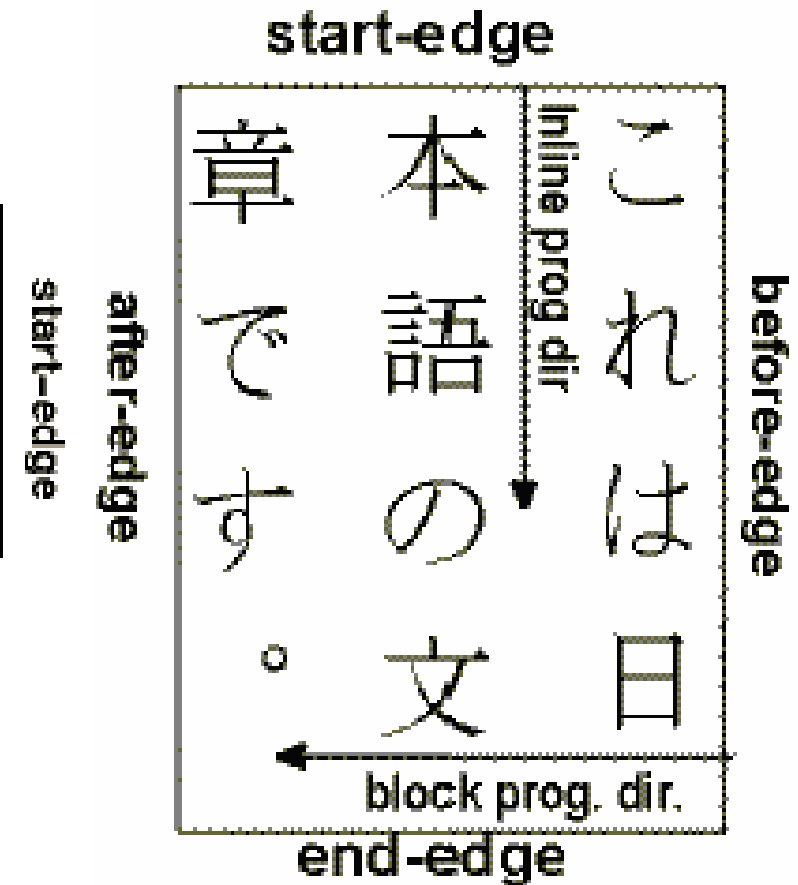
- fo:region-start - начало,
- fo:region-end - конец,
- fo:region-before - пред-область,
- fo:region-after - после-область,
- fo:region-body - основная область (обязательный элемент)
- Простой макет:  
`<fo:region-body/>`
- Области учитывают друг друга с помощью полей (`extent`)

# Пример макета

```
<fo:simple-page-master master-name="rightPage"  
page-height="12cm"  
page-width="12cm"  
margin-left="1cm"  
margin-right="0.5cm"  
margin-top="0.5cm"  
margin-bottom="0.5cm">  
<fo:region-before extent="1cm"/>>  
<fo:region-after extent="1cm"/>  
<fo:region-body  
margin-top="1.1cm"  
margin-bottom="1.1cm" />  
</fo:simple-page-master>
```

# Направления

- Слева-направо (left-to-right), Справа-налево (right-to-left), Сверху-вниз (top-to-bottom)



# Атрибуты макета

- `master-name`: имя макета
- `page-height`: высота страницы
- `page-width`: ширина страницы
- `margin-bottom`, `margin-left`, `margin-right` и `margin-top` или `margin`: отступ
- `writing-mode`: направление текста
- `reference-orientation`: поворот содержимого

# Структура документа

- Документ = набор страниц
- Страница = набор областей (заголовков, основная часть, конец)
- Область = набор блоков (блок, таблица, список)
- Блок = набор блоков | строка
- Строка = набор внутри-строчных элементов (символы, сноски, мат.выражения)

# Последовательность страниц

- Элемент `fo:page-sequence`
- Каждый `fo:page-sequence` связан с мастер страницей через атрибут `master-name`
- Содержимое последовательности:
  - необязательный элемент `fo:title` — титул последовательности
  - необязательные элементы `fo:static-content` — текст, размещаемый на каждой странице
  - Один элемент `fo:flow` — основные данные

- Содержимое fo:flow — набор блоков
- Основной блочный элемент: fo:block (аналог DIV в HTML или display: block в CSS)
- Пример (текст):

<fo:flow>

<fo:block>Южный кукурузный хлеб</fo:block>

<fo:block>1 стакан муки</fo:block>

<fo:block>4 столовых ложки разрыхлителя</fo:block>

...

</fo:flow>



- Шаблон xslt:

```
<xsl:template match="dish|ingredient|directions|story">
```

```
<fo:block><xsl:apply-templates/></fo:block>
```

```
</xsl:template>
```

- Свойства блока задаются через атрибуты
- Большинство названий и значений идентично названиям из CSS
- Пример:

```
<fo:block font-weight="bold">Южный кукурузный  
хлеб</fo:block>
```

- CSS:

```
dish {font-weight: bold}
```

# Пример XSL-FO и CSS

```
dish {  
display: block;  
font-family: Helvetica, Arial, sans serif;  
font-size: 20pt;  
font-weight: bold;  
text-align: center  
}
```

```
<xsl:template match="dish">  
<fo:block font-family="Helvetica, Arial, sans serif" font-size="20pt"  
font-weight="bold" text-align="center">  
<xsl:apply-templates/>  
</fo:block>  
</xsl:template>
```

# Пример

```
<fo:page-sequence master-reference="A4"
  initial-page-number="1" language="en" country="us">
  <fo:static-content flow-name="xsl-region-before">
    <fo:block>The Periodic Table</fo:block>
  </fo:static-content>
  <fo:static-content flow-name="xsl-region-after">
    <fo:block>p. <fo:page-number/></fo:block>
  </fo:static-content>
  <fo:flow flow-name="xsl-region-body">
    <xsl:apply-templates select="//ATOM"/>
  </fo:flow>
</fo:page-sequence>
```

# Размещение потока

- Атрибут **flow-name** — область размещения потока
  - xsl-region-body — основная часть страницы
  - xsl-region-before — верхняя часть страницы
  - xsl-region-after — нижняя часть страницы
  - xsl-region-start — начальная область
  - xsl-region-end — конечная область
- не может быть двух потоков в одной области в одной последовательности!

```
<?xml version="1.0"?>
<fo:root xmlns:fo="http://www.w3.org/1999/XSL/Format">
  <fo:layout-master-set>
    <fo:simple-page-master master-name="only">
      <fo:region-body/>
    </fo:simple-page-master>
  </fo:layout-master-set>
  <fo:page-sequence master-reference="only">
    <fo:flow flow-name="xsl-region-body">
      <fo:block font-size="20pt" font-family="serif" line-height="30pt">
        Hydrogen
      </fo:block>
      <fo:block font-size="20pt" font-family="serif" line-height="30pt" >
        Helium
      </fo:block>
    </fo:flow>
  </fo:page-sequence>
</fo:root>
```

# Статичное содержание

- Элемент fo:static-content
- Атрибуты аналогичны fo:flow
- Должны располагаться перед элементами fo:flow
- Небольшой размер содержимого
- Пример:

```
<fo:page-sequence master-name="A4">  
  <fo:static-content flow-name="xsl-region-before">  
    <fo:block>The Periodic Table</fo:block>  
  </fo:static-content>  
  <fo:flow flow-name="xsl-region-body">  
    <xsl:apply-templates select="//ATOM"/>  
  </fo:flow>  
</fo:page-sequence>
```

# Нумерация страниц

- Необязательные атрибуты fo:page-sequence
- initial-page-number: номер первой страницы (по умолчанию 1)
  - число — номер первой страницы
  - auto — обычная последовательность
  - auto-odd — последовательность нечетных чисел
  - auto-even — последовательность четных чисел
- force-page-count: указание последней страницы
  - auto — последняя страница аналогична значению initial-page-number
  - even/odd — общее число страниц четно/нечетно
  - end-on-even/end-on-odd — последняя страница четна/нечетна
  - no-force — нет ограничений

- `format`: формат числа
  - `letter-value`: символ
  - `country`: код страны по RFC 1766.
  - `language`: код языка по RFC 1766.
  - `grouping-separator`: разделитель группы
  - `grouping-size`: размер группы цифр
- 
- Вставка номера страницы: элемент `fo:page-number`



# Сложные документы

- Страницы с разным оформлением
- Пример: книга (титул, левая и правая страницы)
- Каждое оформление — отдельная мастер страница
- Группировка мастер страниц внутри элемента `fo:page-sequence-master`
- Дочерние элементы:
  - `fo:single-page-master-reference` — одна страница
  - `fo:repeatable-page-master-reference` — повторное использование мастера для всех страниц
  - `fo:repeatable-page-master-alternatives` — различные мастер страницы для различных страниц

# fo:single-page-master-reference

- Ссылка на объявленную мастер страницу

```
<fo:layout-master-set>
```

```
  <fo:simple-page-master master-name="A4"  
    page-width="297mm" page-height="210mm"  
    margin-top="0.5in" margin-bottom="0.5in"  
    margin-left="0.5in" margin-right="0.5in">
```

```
    <fo:region-body/>
```

```
  </fo:simple-page-master>
```

```
  <fo:page-sequence-master master-name="contents">
```

```
    <fo:single-page-master-reference master-reference="A4"/>
```

```
  </fo:page-sequence-master>
```

```
</fo:layout-master-set>
```

- Последовательность из двух страниц

```
<fo:page-sequence-master master-name="contents">
```

```
  <fo:single-page-master-reference master-  
reference="front"/>
```

```
  <fo:single-page-master-reference master-  
reference="back"/>
```

```
</fo:page-sequence-master>
```

- Вторая мастер страница задействуется если переполнится содержимое первой

# fo:repeatable-page-master-reference

- Использование до окончания потока
- атрибут `maximum-repeats` — ограничение числа использований

```
<fo:page-sequence-master master-name="contents">
```

```
  <fo:single-page-master-reference master-  
reference="front"/>
```

```
  <fo:repeatable-page-master-reference master-  
reference="A4" maximum-repeats="10"/>
```

```
</fo:page-sequence-master>
```

# fo:repeatable-page-master-alternatives

- Мастер страница определяется дочерним элементом fo:conditional-page-master-reference
- Атрибуты:
  - master-reference: идентификатор мастер страницы
  - page-position: страница для которой применяется мастер (первая (first), последняя (last), внутренние (rest), все (any))
  - odd-or-even: четные (even), нечетные (odd), все (any) страницы
  - blank-or-not-blank: пустые (blank), непустые (not-blank), все (any)

# Пример

```
<fo:page-sequence-master master-name="contents">  
  <fo:repeatable-page-master-alternatives>  
    <fo:conditional-page-master-reference  
      page-position="first" master-reference="letter_first"/>  
    <fo:conditional-page-master-reference  
      page-position="rest" master-reference="letter"/>  
  </fo:repeatable-page-master-alternatives>  
</fo:page-sequence-master master-reference="contents">
```

# Внутри-строчные форматирующие объекты

- Прямоугольные зоны
- Содержат текст или другие внутри-строчные объекты
- fo:character - символ
- fo:external-graphic — не XML графика
- fo:initial-property-set — форматирование первой строки
- fo:instream-foreign-object — вставка XML не XSL-FO документа (SVG, MathML)
- fo:inline — контейнер для внутри-строчного содержимого
- fo:inline-container - блок-контейнер
- fo:leader — пунктиры и линейки
- fo:page-number — номер страницы

# Пример fo:inline

```
<fo:static-content flow-name="xsl-region-after">  
  <fo:block font-weight="bold" font-size="10pt"  
    font-family="Arial, Helvetica, sans">  
    <fo:inline font-style="italic" text-align="start">  
      The XML Bible  
    </fo:inline>  
    <fo:inline text-align="centered">  
      Page <fo:page-number/>  
    </fo:inline>  
    <fo:inline text-align="right">  
      Глава 18: Форматирующие объекты XSL  
    </fo:inline>  
  </fo:block>  
</fo:static-content>
```



# Табличные форматизирующие объекты

- Более естественное представление чем в CSS
- fo:table-and-caption — контейнер для таблицы и подписи
- fo:table — контейнер таблица (header, body и footer)
- fo:table-caption — подпись к таблице
- fo:table-column — колонка
- fo:table-header — заголовок
- fo:table-footer — окончание
- fo:table-body — тело (row)
- fo:table-row — строка (cell)
- fo:table-cell — ячейка

# Атрибуты табличных элементов

- `table-and-caption` → `caption-side` — расположение описания
- `table-cell` → `number-columns-spanned` и/или `number-rows-spanned` — перекрытие ячеек
- `table` → `table-omit-header-at-break` и `table-omit-footer-at-break` — `false`: верх и низ повторяются на каждой странице, `true` — верх на первой, низ на последней

# Списки

- `fo:list-block` создает список как блоковый элемент
- может быть оформлен буллетами, цифрами, отступами или другим форматированием
- содержит либо серии элементов `fo:list-item` либо пары `fo:list-item-label` + `fo:list-item-body`
- `fo:list-item` содержит `fo:list-item-label` и `fo:list-item-body`
- `fo:list-item-label` — маркер списка
- `fo:list-item-body` — содержимое элемента в виде блоков

# Пример

```
<fo:list-block>
  <fo:list-item-label>
    <fo:block>*</fo:block>
  </fo:list-item-label>
  <fo:list-item-body>
    <fo:block>Actinium</fo:block>
  </fo:list-item-body>
  <fo:list-item-label>
    <fo:block>*</fo:block>
  </fo:list-item-label>
  <fo:list-item-body>
    <fo:block>Aluminum</fo:block>
  </fo:list-item-body>
</fo:list-block>
```

# Атрибуты fo:list-block

- `provisional-label-separation`: расстояние между отметкой пункта списка и телом пункта списка (триплет максимум;минимум;оптимум, например, `2mm;0.5mm;1mm`).
- `provisional-distance-between-starts`: расстояние между начальным краем отметки пункта списка и начальным краем тела пункта списка.

# Вне-строчные форматирующие объекты

- Вне-строчные форматирующие объекты "наплывают" на место имеющихся на странице
- fo:float — плавающий блок
- fo:footnote — сноска
- fo:footnote-body — текст сноски

# Пример сноски

```
<fo:footnote>
```

```
  <fo:inline font-size="smaller" vertical-align="super">*
```

```
</fo:inline>
```

```
<fo:footnote-body font-size="smaller">
```

```
  <fo:inline font-size="smaller" vertical-align="super">
```

```
    *
```

```
</fo:inline>
```

```
  <fo:inline font-style="italic">JavaBeans</fo:inline>,
```

```
  Elliotte Rusty Harold
```

```
  (IDG Books, Foster City, 1998), p. 147
```

```
</fo:footnote-body>
```

```
</fo:footnote>
```

# Пунктиры и линейки

- Линейка - это блок, содержащий горизонтальную линию
- Пунктир - это линия в середине строки, идущая от правой стороны выровненного по левому краю текста до левой стороны какого-то выровненного по правому краю текста на той же строке
- создаются элементом `fo:leader`



# Атрибуты fo:leader

- leader-alignment — выравнивание: reference-area или page (с началом названного объекта), none и inherit.
- leader-length — длина линейки, например, 12pc или 5in.
- leader-pattern — шаблон: space, rule, dots, use-content (символ в содержимом элемента) или inherit
- leader-pattern-width — длина шаблона, например, 2mm или use-font-metrics (естественное расстояние м/у символами).
- rule-style — это свойство может иметь те же значения. что и CSS-свойство border-style; то есть, none, dotted, dashed, solid, double, groove, ridge или inherit.
- rule-thickness — толщина (ширина) линии (по умолчанию - 1pt)

# Пример

```
<fo:static-content flow-name="xsl-region-after">  
<fo:block><fo:leader leader-pattern="rule" leader-  
length="18cm" />  
</fo:block>  
<fo:block>p. <fo:page-number/></fo:block>  
</fo:static-content>
```

# fo:external-graphic

- эквивалент HTML-элемента IMG
- всегда пустой и не содержит дочерних элементов
- Атрибут src содержит URI
- Пример (внутри строчный элемент):

```
<fo:external-graphic  
src="http://www.ibiblio.org/xml/cup.gif"/>
```

- Пример (блочный элемент):

```
<fo:block><fo:external-graphic src="cup.gif"/></fo:block>
```

# fo:instream-foreign-object

- Вставка XML документа
- Распознавание формата и отображение зависит от процессора XSL-FO
- Пример:

```
<fo:instream-foreign-object>
```

```
  <svg xmlns="http://www.w3.org/2000/svg" width="1.5cm"  
height="1cm">
```

```
  <polygon style="fill:#FFCCCC" points="0,31 18,0 36,31"/>
```

```
</svg>
```

```
</fo:instream-foreign-object>
```

# Атрибуты графических объектов

- `content-type` — MIME-тип графики с префиксом "content-type:"
- `height` и `width` — вертикальный и горизонтальный размер зоны
- `content-height` и `content-width` — размеры изображения
- `scaling` — масштабирование пропорциональное `uniform` или непропорциональное `non-uniform`
- `scaling-method` — алгоритм масштабирования

# ССЫЛКИ

- создание ссылок в стиле гиперссылок HTML с помощью элемента `fo:basic-link`
- Атрибуты:
  - `external-destination` — удаленная цель (URI)
  - `internal-destination` — внутренний узел текущего документа (ID)
  - `indicate-destination` — выделять объект или нет (`true` или `false`)
  - `show-destination` — `replace`: замена документа, `new`: новое окно
  - `destination-placement-offset` — сдвиг объекта-цели вниз
  - `target-presentation-context` — отображение части объекта-цели
  - `target-processing-context` — базовый адрес (URI)
  - `target-style-sheet` — таблица стилей для документа-цели