



Введение в Модель C4 и Structurizr Часть 1

Д. Макаренков, к.т.н.

<https://dmpsy.club>

C4Model => Structurizr



Целевая аудитория

Архитекторы ПО, желающие применять современный подход иерархической модели C4 с использованием Structurizr

Магическая формула:

C4 Model => Structurizr (DSL)



План работы / Agenda

1	C4 model for visualising SW architecture	Модель C4 - средство визуализации архитектуры ПО
2	Basic Principles of C4 model	Базовые принципы модели C4
3	Structurizr, the C4 model implementation	Structurizr - основная реализация модели C4
4	Intro tutorial on Structurizr / DSL	Вводный урок по Structurizr DSL
5	Useful links	Полезные ссылки



Базовые принципы Модели С4

1. Четырехуровневая иерархическая структура: от общего к частному, от систем ПО к коду
2. Единая модель для всех уровней
3. “Диаграмма в виде кода” - универсально и понятно разработчикам, язык DSL (Domain Specific Language)
4. Одна модель - несколько представлений (views)

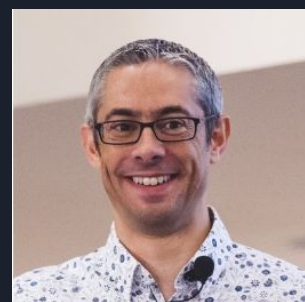
Структура и преимущества Модели C4

Модель C4 для визуализации архитектуры ПО (автор - Саймон Браун):

1. Набор иерархических абстракций (программные системы, контейнеры, компоненты и код)
2. Набор иерархических диаграмм (контекста системы, контейнеров, компонентов и кода)
3. Дополнительные диаграммы: ландшафта, развертывания, динамические

Модель не зависит от нотации и используемого инструментария

Модель C4 — простой в освоении, интуитивно понятный подход к визуализации архитектуры ПО. Взаимосвязанные диаграммы архитектуры ПО упрощают взаимодействие внутри и вне групп разработки ПО, введение в курс дела новых сотрудников, комплексную оценку архитектурного решения, идентификацию рисков, моделирование уязвимостей и т. д.



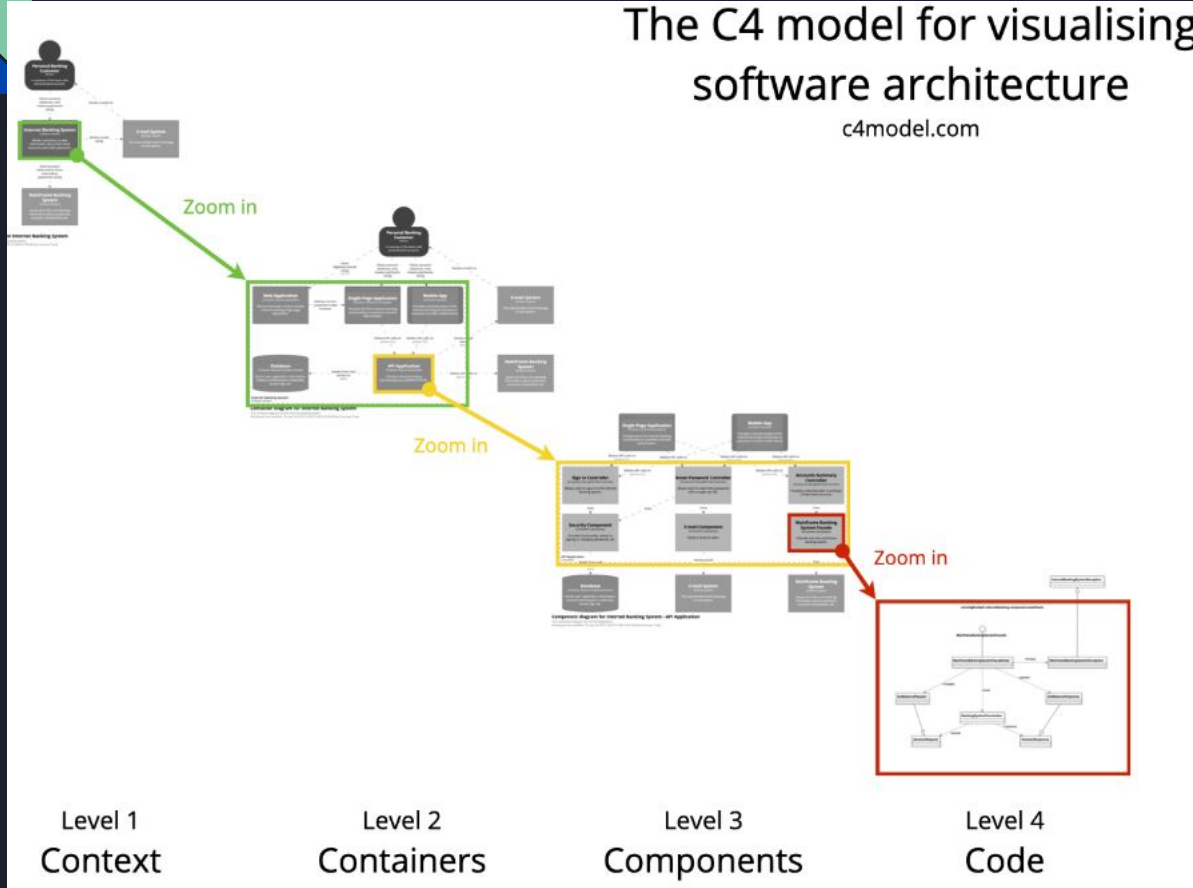
Иерархия Модели C4 наглядно

The C4 model for visualising software architecture

c4model.com

Удобно представлять себе эти четыре уровня как четыре увеличительных окуляра микроскопа

Источник:
<https://c4model.com/images/c4-overview.png>



Вводный урок по Structurizr. Диаграмма контекста (C1)

1	Use Demo page	Откройте https://structurizr.com/dsl
2	Сору / Paste workspace from workspace.dsl	<p>Скопируйте workspace в окно Demo page:</p> <pre>workspace "Name" "Description" { model { u = person "User" ss = softwareSystem "Software System" u -> ss "Uses" "HTTP:80" } views { #system context view systemContext ss "Diagram1" { #include all software systems interacting with this one include * #place the elements left-to-right automatically autolayout lr } } }</pre>

Вводный урок по Structurizr. Диаграмма контекста (2)

3 Click Render and see the context (C1) diagram

Нажмите Render и посмотрите диаграмму контекста (C1) в правом окне

The screenshot displays the Structurizr web interface. On the left, a code editor shows the Structurizr DSL for a workspace named "Software System". The DSL defines a model with a person "User" and a software system "Software System", and a relationship where the user uses the software system. It also defines a system context view named "Diagram1" that includes all software systems and uses an automatic layout.

```
1 workspace "Name" "Description" {
2
3   model {
4     u = person "User"
5     ss = softwareSystem "Software System"
6
7     u -> ss "Uses" "HTTP:80"
8   }
9
10  views {
11    #system context view
12    systemContext ss "Diagram1" {
13      #include all software systems interacting with this one
14      include *
15      #place the elements left-to-right automatically
16      autolayout lr
17    }
18  }
19 }
20
```

On the right, the rendered diagram is shown. It features two rectangular nodes: "User" (labeled as a person) on the left and "Software System" (labeled as a software system) on the right. A directed relationship arrow points from the User node to the Software System node, labeled "Uses" with the identifier "[111]P:80".

At the bottom of the diagram, there is a timestamp: "[System Context] Software System" followed by "суббота, 8 марта 2025 г. в 09:53 Москва, стандартное время".

Вводный урок по Structurizr. Диаграмма контекста (3)

4

**Export your diagram to
*.png**

**Нажмите кнопку Export в правом нижнем углу
диаграммы, затем Export current diagram**

The screenshot shows the Structurizr web interface. At the top, there are several buttons for exporting diagrams: "Structurizr Diagram" (highlighted in blue), "Structurizr Graph", "Export PlantUML", "Export C4-PlantUML", "Export Mermaid", "Export DOT", "Export WebSequenceDiagrams", and "Export Ilograph". Below these buttons is a dropdown menu showing "[System Context] Software System (#Diagram1)".

In the foreground, a dialog box titled "Export to PNG" is open. It contains the following options:

- Include diagram title, description, and metadata
- Crop diagrams
- Include animation steps
- Automatically download

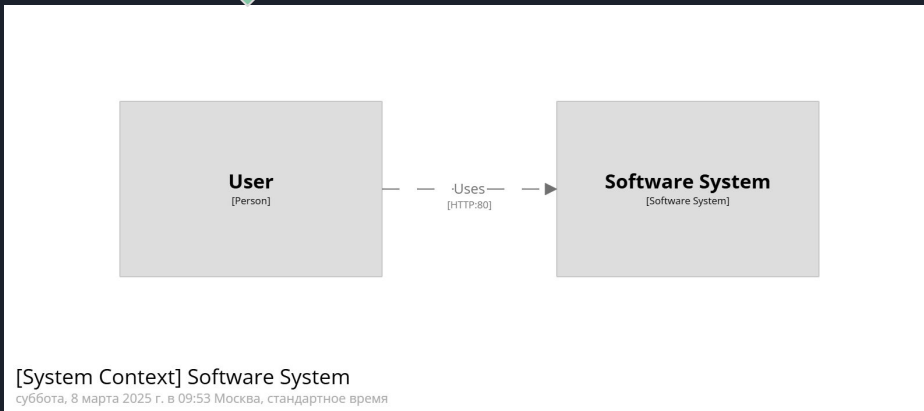
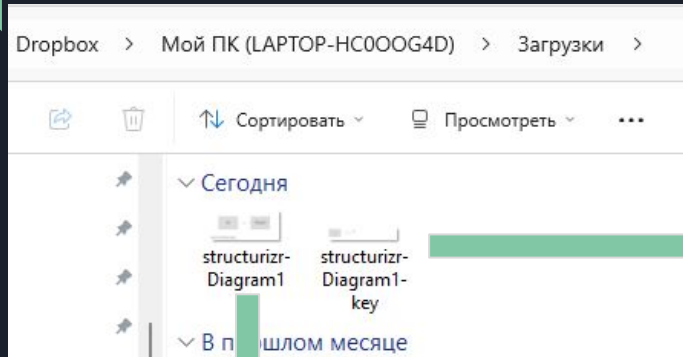
At the bottom of the dialog box is a button labeled "Export current diagram".

Вводный урок по Structurizr. Диаграмма контекста (4)

4

Save and open your diagram locally

Откройте свою диаграмму, сохраненную на PC



Вводный урок по Structurizr. Диаграмма контейнеров (C2)

0	To do: Nest containers into the software system and display the diagram	Задача: <ol style="list-style-type: none">1. Определить контейнеры (приложение и хранилище данных) - составные части нашей программной системы - поместив их в ее область действия (внутри фигурных скобок)2. Определить соответствующее представление (view) и вывести диаграмму C2
1	Use Demo page	Откройте https://structurizr.com/dsl

Вводный урок по Structurizr. Диаграмма контейнеров (2)

2

**Copy / Paste
workspace from
workspace.dsl**

Скопируйте workspace в окно Demo page:

```
workspace "Name" "Description" {  
  
    #this allows one to refer to those containers  
    #via their fully qualified identifier.  
    !identifiers hierarchical  
  
    model {  
        u = person "User"  
        ss = softwareSystem "Software System" {  
            wa = container "Web Application"  
            db = container "Database Schema" {  
                tags "Database"  
            }  
        }  
    }  
  
    u -> ss "Uses" "HTTP:80"  
    u -> ss.wa "Uses" "HTTP:80"  
    ss.wa -> ss.db "Reads from and writes to"  
}
```

См. след. слайд

Вводный урок по Structurizr. Диаграмма контейнеров (3)

2

Copy / Paste
workspace from
workspace.dsl

Окончание кода (создаем два представления, C1 и C2):

```
views {
  #system context view
  systemContext ss "Diagram1" {
    #include all software systems interacting with this one
    include *
    #place the elements left-to-right automatically
    autolayout lr
  }

  #container view
  container ss "Diagram2" {
    #include all the containers and people and software systems,
    #interacting with them
    include *
    #place the elements left-to-right automatically
    autolayout lr
  }
}
```

Вводный урок по Structurizr. Диаграмма контейнеров (4)

3

Click Render and see the container (C2) diagram

Нажмите Render и выберите из списка диаграмму контейнеров (C2) в правом окне

The screenshot displays the Structurizr web application interface. At the top, there is a menu with several options: "Structurizr Diagram" (highlighted in dark blue), "Structurizr Graph" (light blue), "Export PlantUML", "Export C4-PlantUML", "Export Mermaid", "Export DOT", "Export WebSequenceDiagrams", and "Export Ilograph". Below the menu is a dropdown menu showing "[Container] Software System (#Diagram2)". The main area contains a container diagram with three components: "User (Person)", "Web Application (Container)", and "Database Schema (Container)". The "Web Application" and "Database Schema" components are enclosed in a dashed-line box labeled "Software System (SoftwareSystem)". An arrow labeled "Uses (177:00)" points from the User to the Web Application, and another arrow labeled "Reads from and writes to" points from the Web Application to the Database Schema. At the bottom left, the text "[Container] Software System" is visible, along with a timestamp: "воскресенье, 9 марта 2025 г. в 12:07 Москва, стандартное время". At the bottom right, there is a toolbar with icons for zooming, rendering, and other actions.

Вводный урок по Structurizr. Диаграмма контейнеров (5)

4

Export your container diagram to *.png

Нажмите кнопку Export в правом нижнем углу диаграммы, затем Export current diagram

The screenshot shows the Structurizr web interface. At the top, there is a navigation bar with buttons for 'Structurizr Diagram', 'Structurizr Graph', 'Export PlantUML', 'Export C4-PlantUML', 'Export Mermaid', 'Export DOT', 'Export WebSequenceDiagrams', and 'Export Ilograph'. Below the navigation bar is a dropdown menu showing '[Container] Software System (#Diagram2)'. The main content area is titled 'Export to PNG' and contains the following options:

- Include diagram title, description, and metadata
- Crop diagrams
- Include animation steps
- Automatically download

At the bottom of the dialog is a button labeled 'Export current diagram'.

Вводный урок по Structurizr. Диаграмма контейнеров (6)

4

Save and open your container diagram locally

Откройте свою диаграмму, сохраненную на PC



[Container] Software System

воскресенье, 9 марта 2025 г. в 12:33 Москва, стандартное время

Вводный урок по Structurizr. Неявные (implicit) отношения

0	To do: Exclude an extra upper-level relationship and display the diagrams	Задача: <ol style="list-style-type: none">1. Убрать излишнее верхнеуровневое отношение из предыдущего примера, которое уже задано неявно2. Вывести диаграммы C1 и C2 и убедиться, что зависимости не нарушились
1	Use Demo page	Откройте https://structurizr.com/dsl

Вводный урок по Structurizr. Неявные (implicit) отношения (2)

2

Copy / Paste
workspace from
workspace.dsl

Закомментируйте излишнее отношение в workspace из
предыдущего примера в окне Demo page:

```
workspace "Name" "Description" {  
  
    #this allows one to refer to those containers  
    #via their fully qualified identifier.  
    !identifiers hierarchical  
  
    model {  
        u = person "User"  
        ss = softwareSystem "Software System" {  
            wa = container "Web Application"  
            db = container "Database Schema" {  
                tags "Database"  
            }  
        }  
    }  
  
    # u -> ss "Uses" "HTTP:80"  
    u -> ss.wa "Uses" "HTTP:80"  
    ss.wa -> ss.db "Reads from and writes to"  
}
```

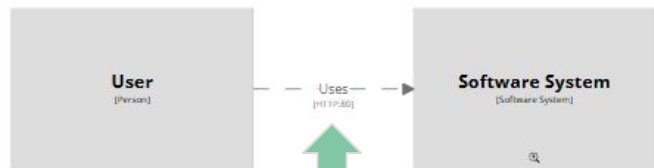
Вводный урок по Structurizr. Неявные (implicit) отношения (3)

3

Click Render and see the (C1 and C2) diagrams

Нажмите Render и выведите диаграммы C1 и C2 в правом окне

[System Context] Software System (#Diagram1) ▾



Implicit Relationship:
u -> ss "Uses" "HTTP:80"

[Container] Software System (#Diagram2) ▾



Explicit Relationship:
u -> ss.wa "Uses" "HTTP:80"

Вводный урок по Structurizr. Выражения DSL в представлениях

0	To do: Use DSL expressions in views	Задача: <ol style="list-style-type: none">1. Использовать выражения языка DSL для конфигурирования представлений (views)2. Вывести соответствующие диаграммы
1	Use Demo page	Откройте https://structurizr.com/dsl

Вводный урок по Structurizr. Выражения DSL в представлениях (2)

2

**Copy / Paste
several
views**

Изучите и скопируйте следующее представление в конец области views {...} предыдущего примера:

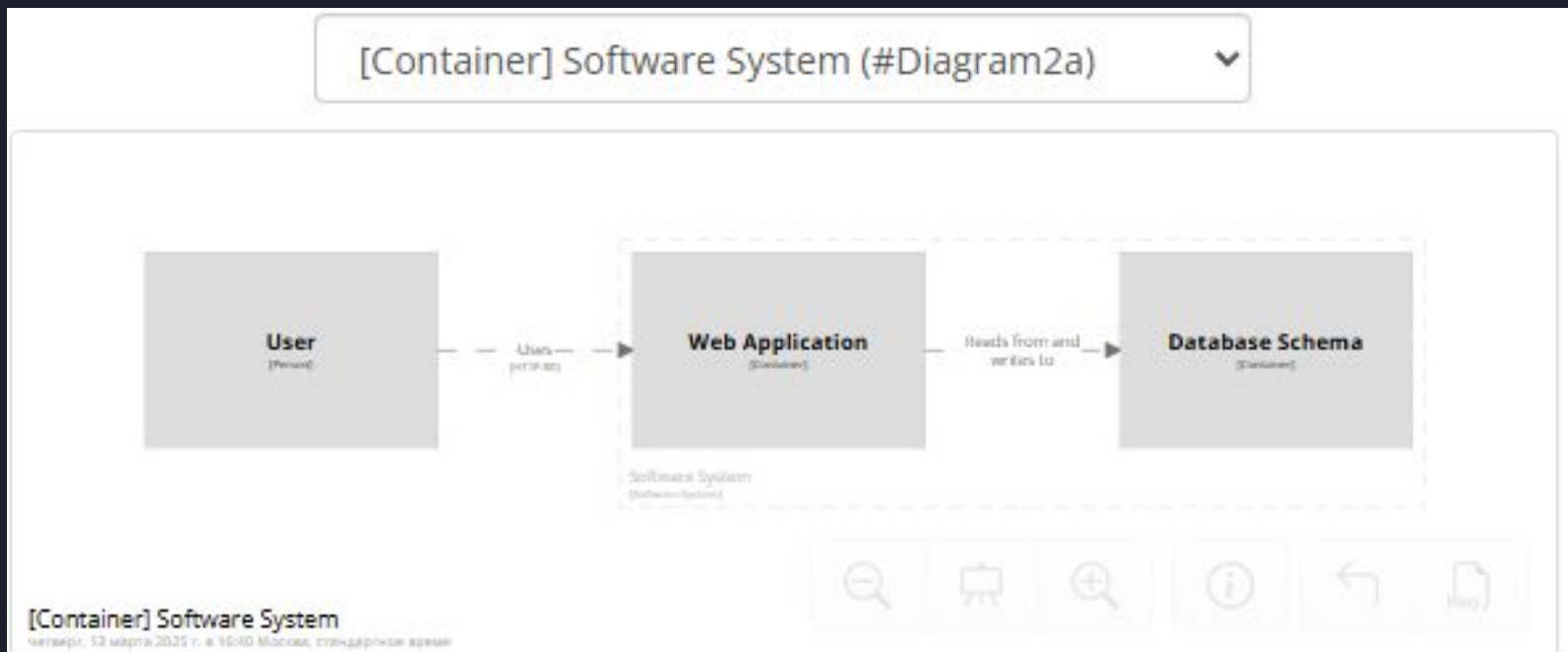
```
#container view  
  
container ss "Diagram2a" {  
    #include the user, web app, and database explicitly  
    include u ss.wa ss.db  
  
    #place the elements left-to-right automatically  
    autolayout lr  
  
}
```

Вводный урок по Structurizr. Выражения DSL в представлениях (3)

3

Click Render and see the diagram

Нажмите Render и выведите диаграмму



Вводный урок по Structurizr. Выражения DSL в представлениях (4)

4

**Copy / Paste
several
views**

**Изучите и скопируйте следующее представление в конец области
views {...} предыдущего примера:**

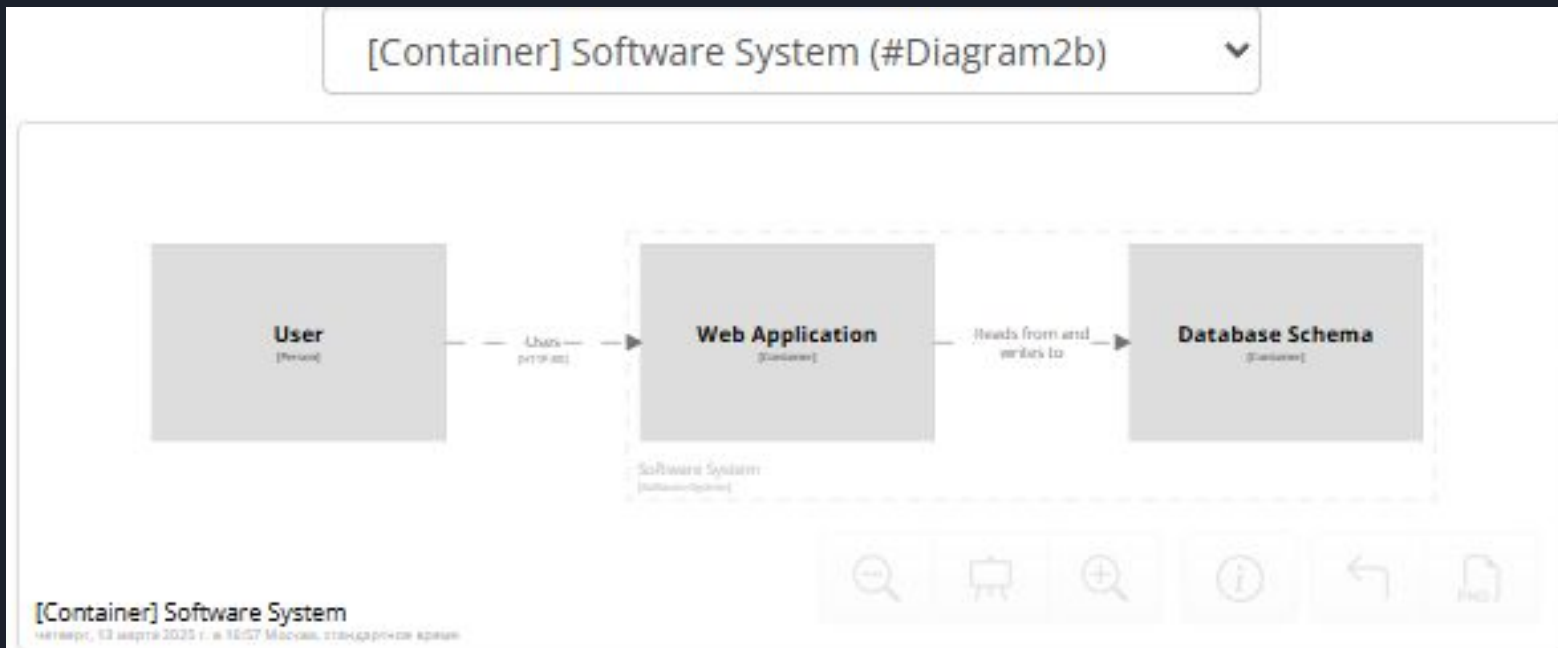
```
#container view

    container ss "Diagram2b" {
        #include the user, web app, and database explicitly
        #on separate lines just for better readability
        include u
        include ss.wa ss.db
        #place the elements left-to-right automatically
        autolayout lr
    }
```

Вводный урок по Structurizr. Выражения DSL в представлениях (5)

5 Click Render and see the diagram

Нажмите Render и выведите диаграмму



Вводный урок по Structurizr. Выражения DSL в представлениях (6)

6

**Copy / Paste
several
views**

Изучите и скопируйте следующее представление в конец области views {...} предыдущего примера:

```
#container view
```

```
    container ss "Diagram2c" {
```

```
        #include the web app, and
```

```
        #and its inbound and outbound dependencies
```

```
        include "->ss.wa->"
```

```
        #place the elements left-to-right automatically
```

```
        autolayout lr
```

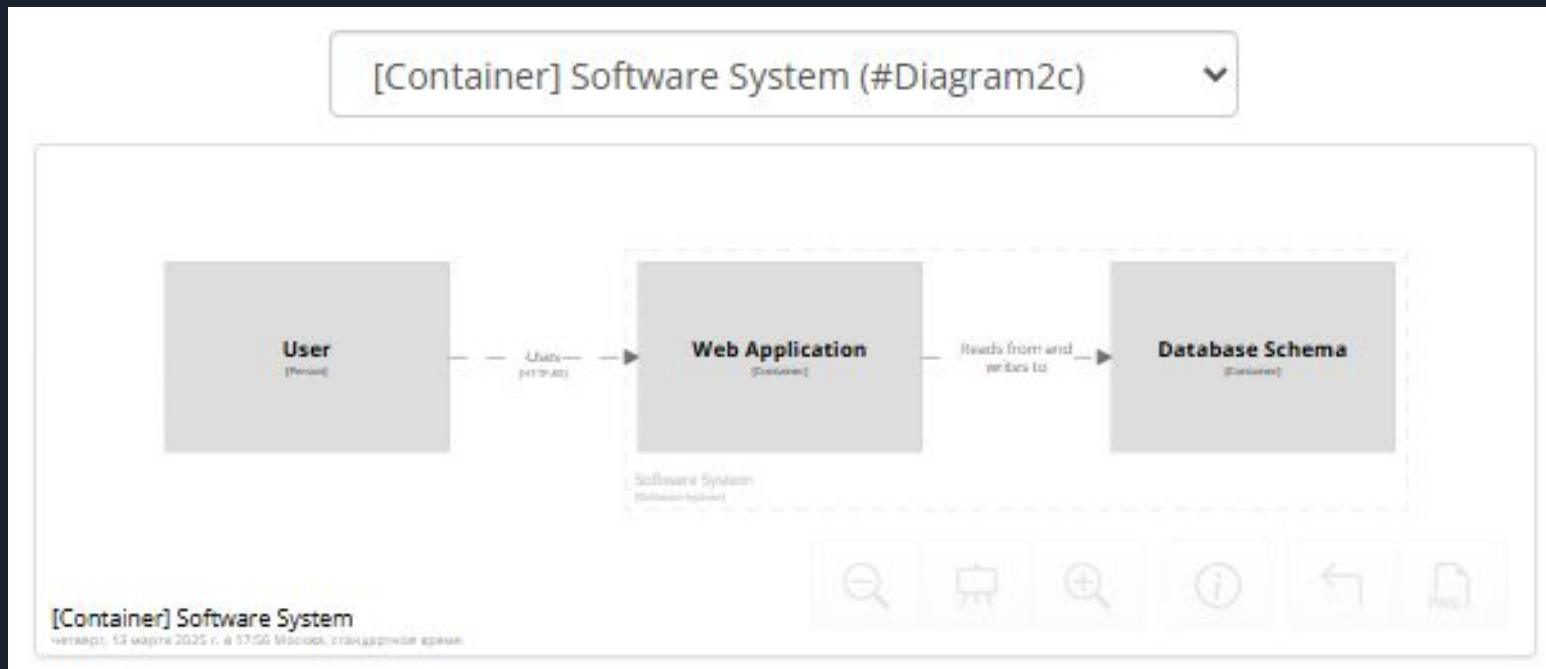
```
    }
```

Вводный урок по Structurizr. Выражения DSL в представлениях (7)

7

Click Render and see the diagram

Нажмите Render и выведите диаграмму



Вводный урок по Structurizr. Выражения DSL в представлениях (7)

7

**Copy / Paste
several
views**

**Изучите и скопируйте следующее представление в конец области
views {...} предыдущего примера:**

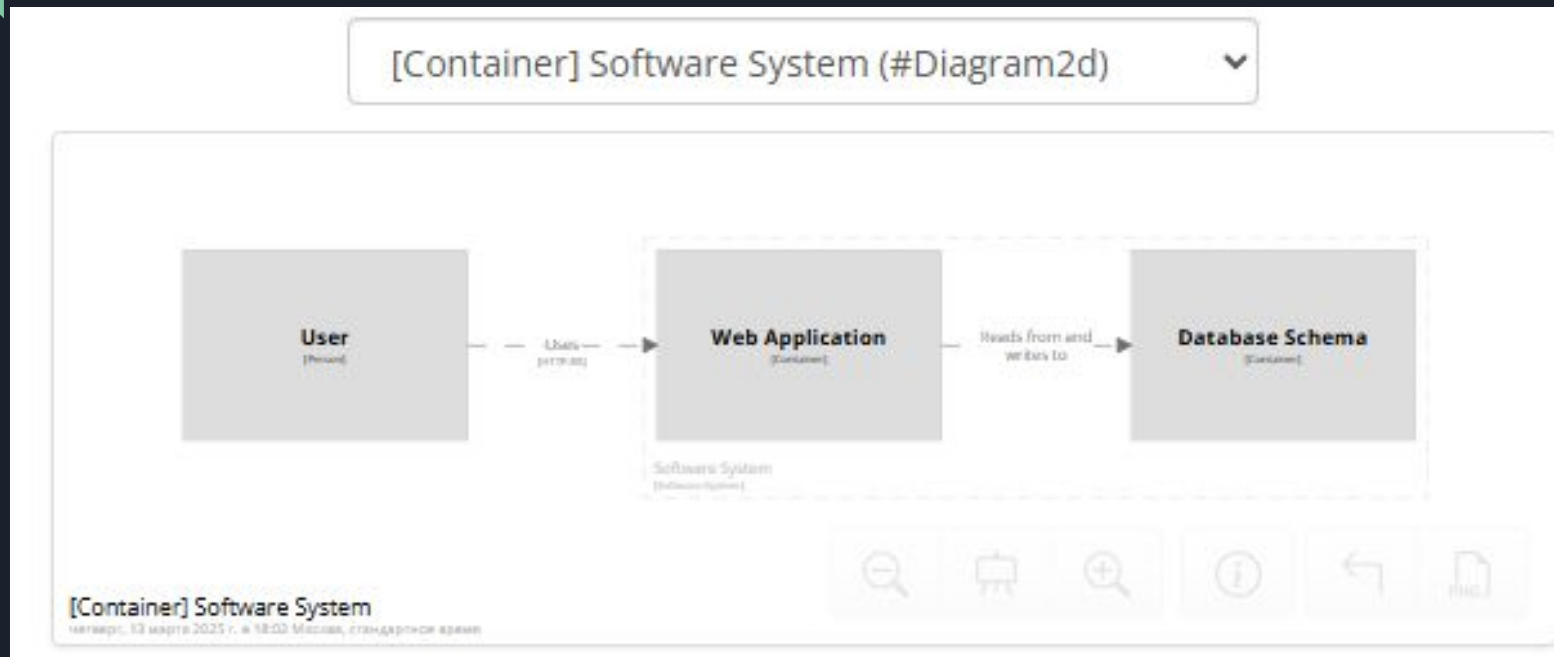
```
#container view
    container ss "Diagram2d" {
        #include elements of type container
        #and their inbound and outbound dependencies
        include "->element.type==container->"
        #place the elements left-to-right automatically
        autolayout lr
    }
```

Вводный урок по Structurizr. Выражения DSL в представлениях (8)

8

Click Render and see the diagram

Нажмите Render и выведите диаграмму



Вводный урок по Structurizr. Выражения DSL в представлениях (9)

9

**Copy / Paste
several
views**

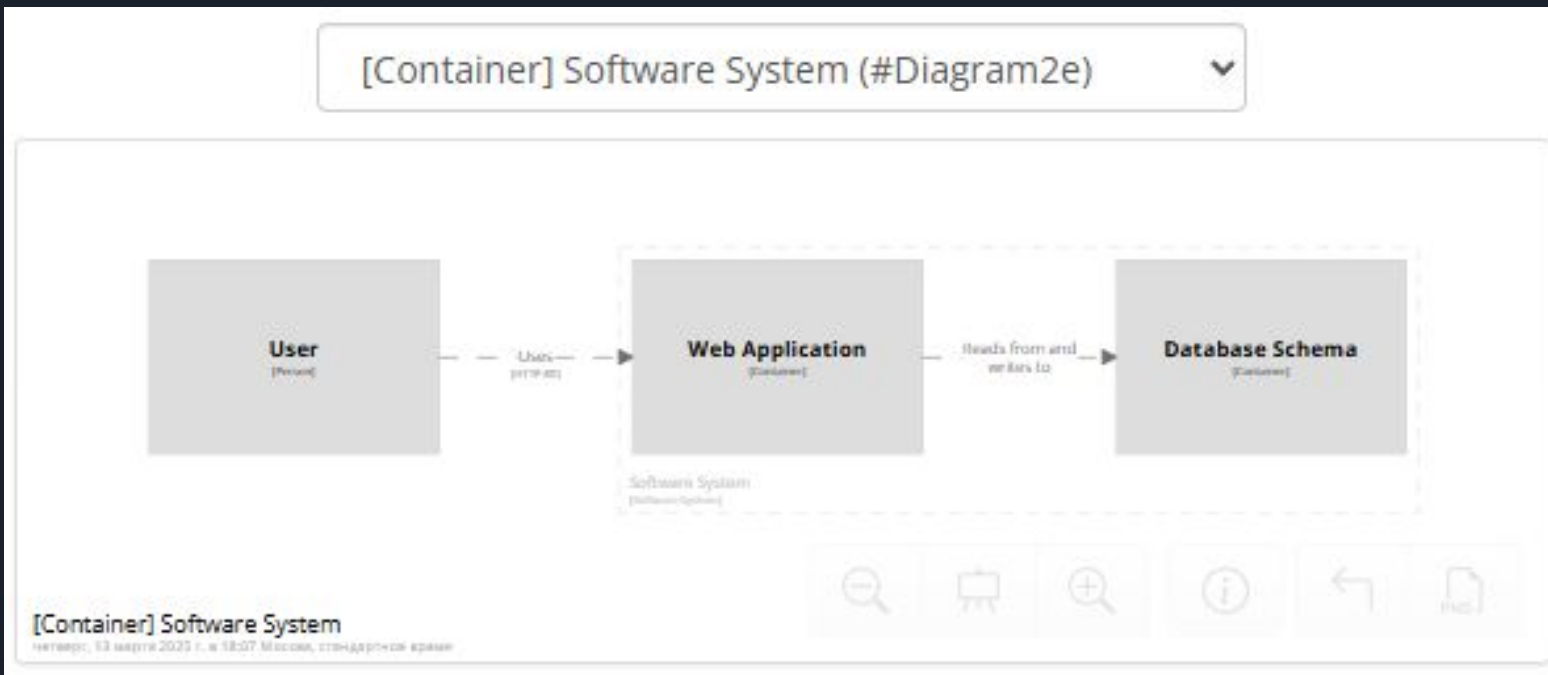
**Изучите и скопируйте следующее представление в конец области
views {...} предыдущего примера:**

```
#container view
    container ss "Diagram2e" {
        #include children of the (ss) parent element
        #and their inbound and outbound dependencies
        include "->element.parent==ss->"
        #place the elements left-to-right automatically
        autolayout lr
    }
```

Вводный урок по Structurizr. Выражения DSL в представлениях (10)

10 Click Render and see the diagram

Нажмите Render и выведите диаграмму



Вводный урок по Structurizr. Применение стилей

0	To do: Use DSL expressions to style views	Задача: <ol style="list-style-type: none">1. Использовать выражения языка DSL для “стилизации” представлений (views)2. Вывести соответствующие диаграммы
1	Use Demo page	Откройте https://structurizr.com/dsl

Вводный урок по Structurizr. Применение стилей (2)

2

**Copy / Paste
styles**

**Изучите и скопируйте следующие стили в конец области views {...}
предыдущего примера:**

```
styles {  
    #set the foreground colour of all elements to white  
    element "Element" {  
        color white  
    }  
  
    element "Container" {  
        #set the background color of all containers to lighter green  
        background #55aa55  
    }  
  
    element "Person" {  
        #set the background color of all people to darker green  
        background #116611  
        #set the shape of all people to a person shape  
        shape person  
    }  
}
```

см. следующий слайд

Вводный урок по Structurizr. Применение стилей (3)

2

**Copy / Paste
styles**

**Изучите и скопируйте следующие стили в конец области views {...}
предыдущего примера:**

```
element "Database" {  
    #set the user-defined Database element  
    #to be of the cylinder shape  
    shape cylinder  
}
```

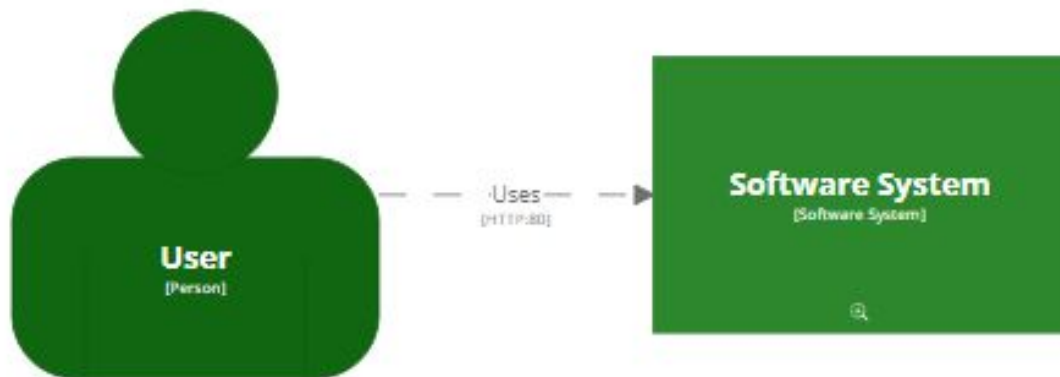
```
element "Software System" {  
    #set the background color of all software systems to green  
    background #2D882D  
}  
}
```

Вводный урок по Structurizr. Применение стилей (4)

3

Click Render and see the (C1) diagram

Нажмите Render и выведите диаграмму C1



[System Context] Software System

четверг, 13 марта 2025 г. в 19:34 Москва, стандартное время



Вводный урок по Structurizr. Применение стилей (5)

3 Click Render and see the (C2) diagram

Нажмите Render и выведите диаграмму C2



[Container] Software System

четверг, 13 марта 2025 г. в 19:38 Москва, стандартное время





Введение в Модель C4 и Structurizr Часть 2

Д. Макаренков, к.т.н.

<https://dmpsy.club>

C4Model => Structurizr

Вводный урок по Structurizr. Диаграмма компонентов (C3)

0	To do: Nest components into containers and display the (C3) diagrams	Задача: <ol style="list-style-type: none">1. Определить компоненты, поместив их в область действия двух контейнеров (внутри фигурных скобок)2. Определить соответствующие представления (view) и вывести диаграммы C3
1	Use Demo page	Откройте https://structurizr.com/dsl

Вводный урок по Structurizr. Диаграмма компонентов (2)

2

Сору / Paste workspace from workspace.dsl

Скопируйте workspace.dsl из https://dmpsy.club/references/Structurizr/006_Components/workspace.dsl в окно Demo page и проанализируйте измененную модель:

```
model {  
    u = person "User"  
    ss = softwareSystem "Software System" {  
        wa = container "Web Application" {  
            main_page = component "Main Page"  
            login_page = component "Login Page"  
            registration_page = component "Registration Page"  
            user_account_page = component "User Personal Page"  
            main_page -> login_page "Contains"  
            main_page -> registration_page "Contains"  
            login_page -> user_account_page "Opens"  
            registration_page -> user_account_page "Opens"  
        }  
    }  
}
```

См. след. слайд

Вводный урок по Structurizr. Диаграмма компонентов (3)

2

**Copy / Paste
workspace
from
workspace.dsl**

...проанализируйте измененную модель:

```
model {  
  ...  
  db = container "Database Schema" {  
    #tag this element for styling specified below  
    tags "Database"  
    user = component "Users Table"  
    orders = component "Orders Table"  
    user_orders_v = component "User Orders View"  
    user_orders_v -> user "Gets a current user"  
    user_orders_v -> orders "Retrieves user orders"  
  }  
  ...  
}
```

См. след. слайд

Вводный урок по Structurizr. Диаграмма компонентов (4)

2

**Copy / Paste
workspace
from
workspace.dsl**

...проанализируйте измененную модель:

```
model {
    ...
    wa -> db "Reads from and writes to"
    wa.registration_page -> db.user "Adds new user
credentials"
    wa.login_page -> db.user "Retrieves user credentials"
    wa.user_account_page -> db.user_orders_v "Retrieves
user orders"
    wa.user_account_page -> db.user "Manages user
credentials"
    wa.user_account_page -> db.orders "Manages user orders"
}
u -> ss.wa.main_page "Uses" "HTTP:80"
}
```


Вводный урок по Structurizr. Диаграмма компонентов (5)

2

Copy /
Paste
workspace
from
workspace.
dsl

...проанализируйте добавленные представления:

```
#component view
    component ss.wa "WAComponents" {
        #include all components
        #and their inbound and outbound dependencies
        include *
        #place the elements left-to-right automatically
        autolayout lr
    }
#component view
    component ss.db "DBComponents" {
        #include all components
        #and their inbound and outbound dependencies
        include *
        #place the elements left-to-right automatically
        autolayout lr
    }
```

Вводный урок по Structurizr. Диаграмма компонентов (6)

2

Copy /
Paste
workspace
from
workspace.
dsl

...И СТИЛЬ КОМПОНЕНТОВ:

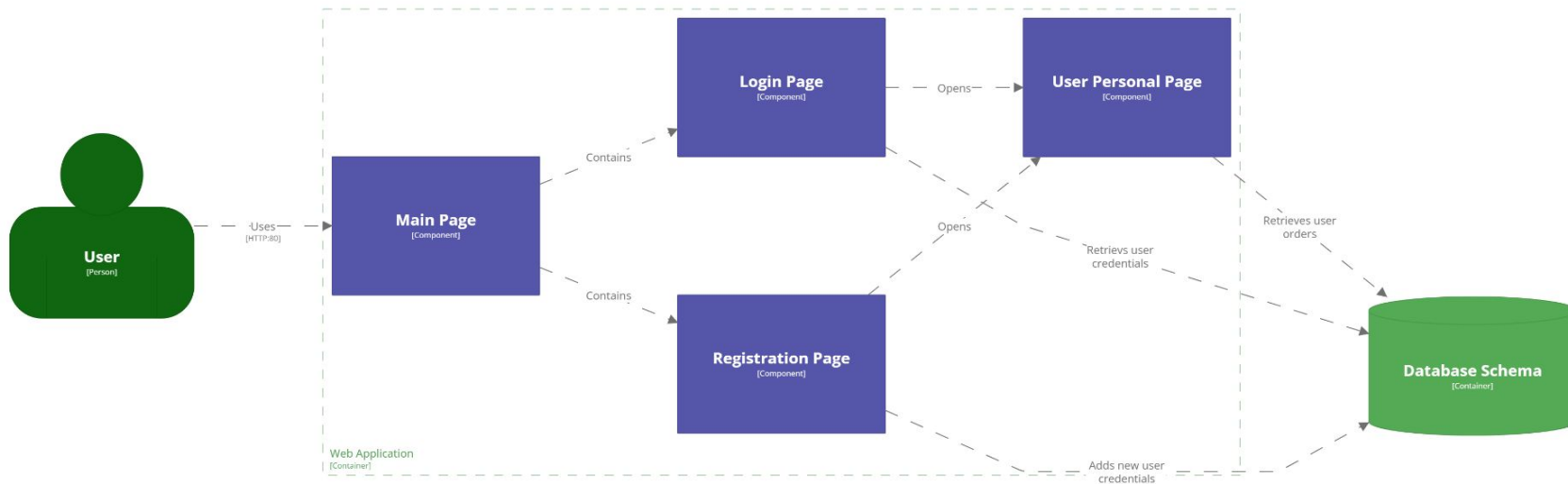
```
element "Component" {  
#set the background color of all components to lighter blue  
    background #5555aa  
}
```

Вводный урок по Structurizr. Диаграмма компонентов (7)

3

Click Render and see the container (C3) diagram

Нажмите Render и выберите из списка 1ю диаграмму контейнеров (C3) в правом окне

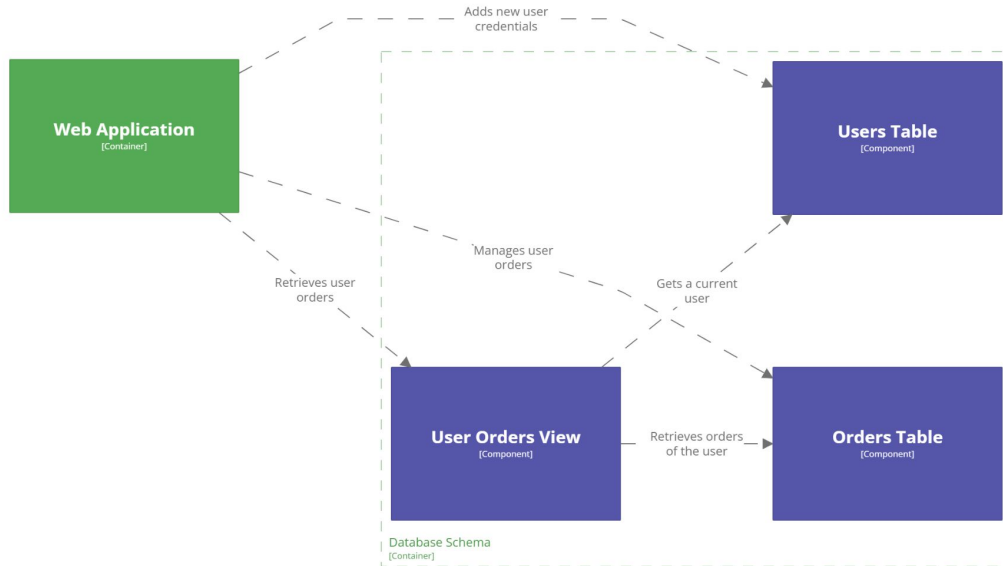


Вводный урок по Structurizr. Диаграмма компонентов (8)

3

Click Render and see the container (C3) diagram

Нажмите Render и выберите из списка 2ю диаграмму контейнеров (C3) в правом окне



Вводный урок по Structurizr. Диаграмма кода (C4)

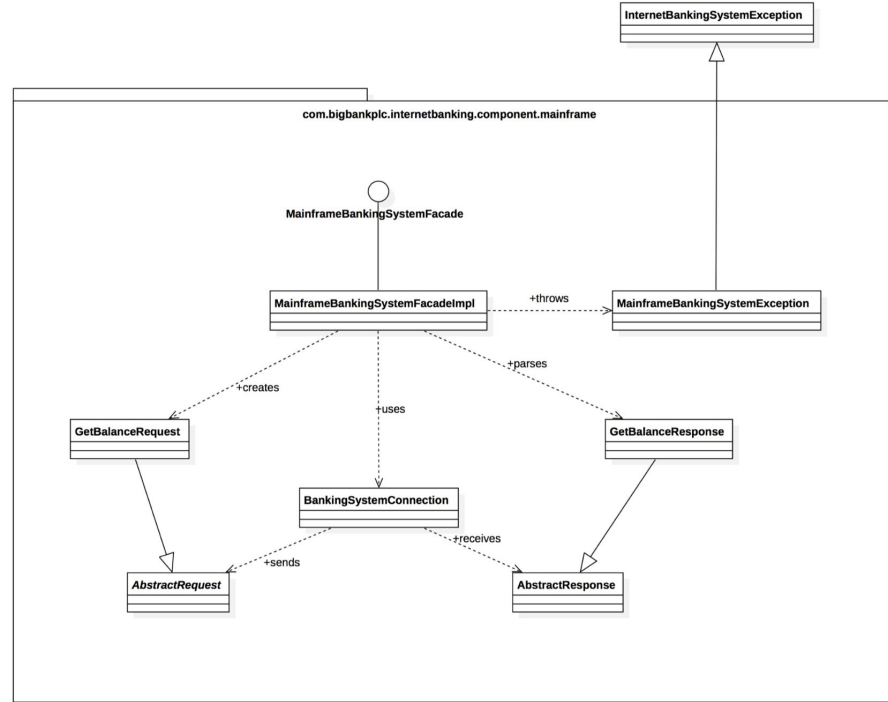
0	To do: Understand what (C4) diagram is and how to use it properly	<p>Диаграмма кода - последний в иерархии, опциональный уровень детализации, рекомендованный исключительно для наиболее сложных компонентов системы</p> <p>Схема представляет собой элементы кода (классы, интерфейсы, объекты, функции, таблицы базы данных и т. д.) в пределах (scope) одного компонента</p>
---	--	--

Вводный урок по Structurizr. Диаграмма кода (2)

1

Example:

<https://static.structurizr.com/workspace/36141/diagrams/MainframeBankingSystemFacade.png>



Вводный урок по Structurizr. Диаграмма кода (3)

3

Embed the (C4) diagram into the views {...} section of workspace.dsl

or copy/paste workspace.dsl from

https://dmpsy.club/references/Structurizr/007_Code/workspace.dsl

В предыдущем примере объявите компонент внутри контейнера ss.wa

```
mainframeBankingSystemFacade = component "Mainframe Banking System Facade" "A facade onto the mainframe banking system." "Spring Bean"
```

и поместите диаграмму C4 в раздел views {...}:

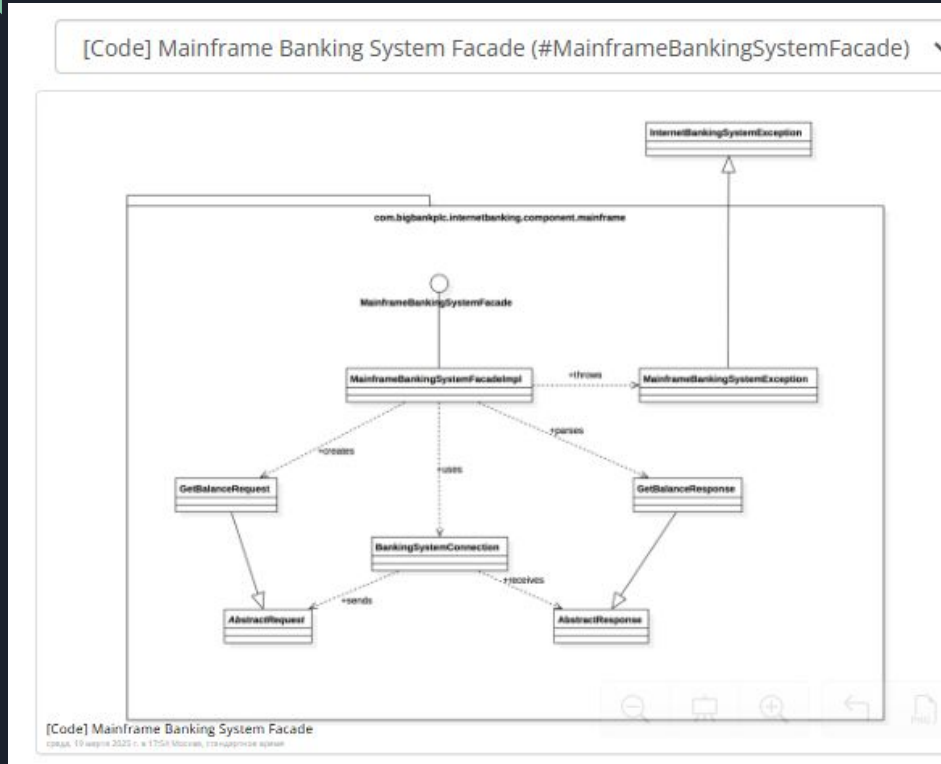
```
image ss.wa.mainframeBankingSystemFacade "MainframeBankingSystemFacade" {  
    image  
    https://raw.githubusercontent.com/structurizr/examples/main/dsl/big-bank-plc/internet-banking-system/mainframe-banking-system-facade.png  
    title "[Code] Mainframe Banking System Facade"  
}
```

Вводный урок по Structurizr. Диаграмма кода (4)

3

Click Render and see the code (C4) diagram

Нажмите Render и выберите из списка диаграмму кода (C4) в правом окне



Вводный урок по Structurizr. Диаграмма ландшафта (“C0”)

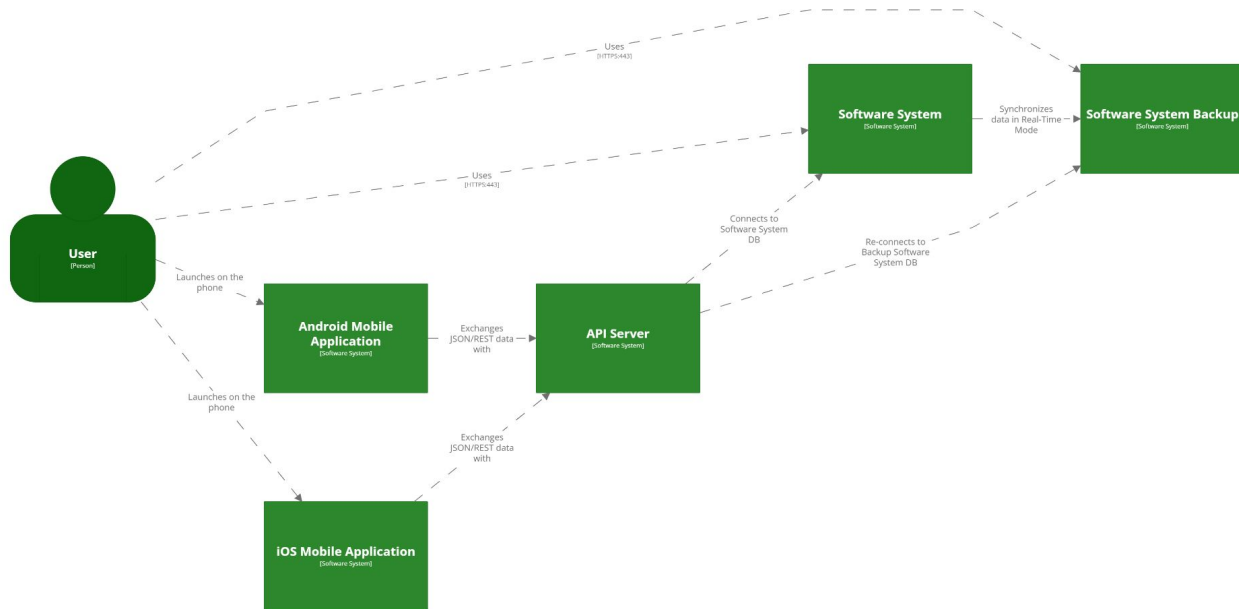
0	To do: Understand what (“C0”) diagram is and how to use it properly	<p>Диаграмма ландшафта показывает окружение нашей системы: пользователей и системы ПО, взаимодействующие с ней</p> <p>По сути, это “мост” от контекста конкретной системы к архитектуре компании, нужный и важный для корпоративных системных архитекторов</p> <p>С точки зрения иерархии может быть обозначена как “C0”, хотя, по своей сути, представляет собой расширенную до нескольких систем диаграмму контекста C1</p>
1	Use Demo page	Откройте https://structurizr.com/dsl

Вводный урок по Structurizr. Диаграмма ландшафта (2)

2

Copy / Paste workspace from workspace.dsl

Скопируйте workspace.dsl из https://dmpsy.club/references/Structurizr/008_SystemLandscape/worksapce.dsl и выведите диаграмму ландшафта - эскиз архитектуры Интернет-магазина, созданный на основе предыдущих примеров:



Вводный урок по Structurizr. Диаграмма ландшафта (3)

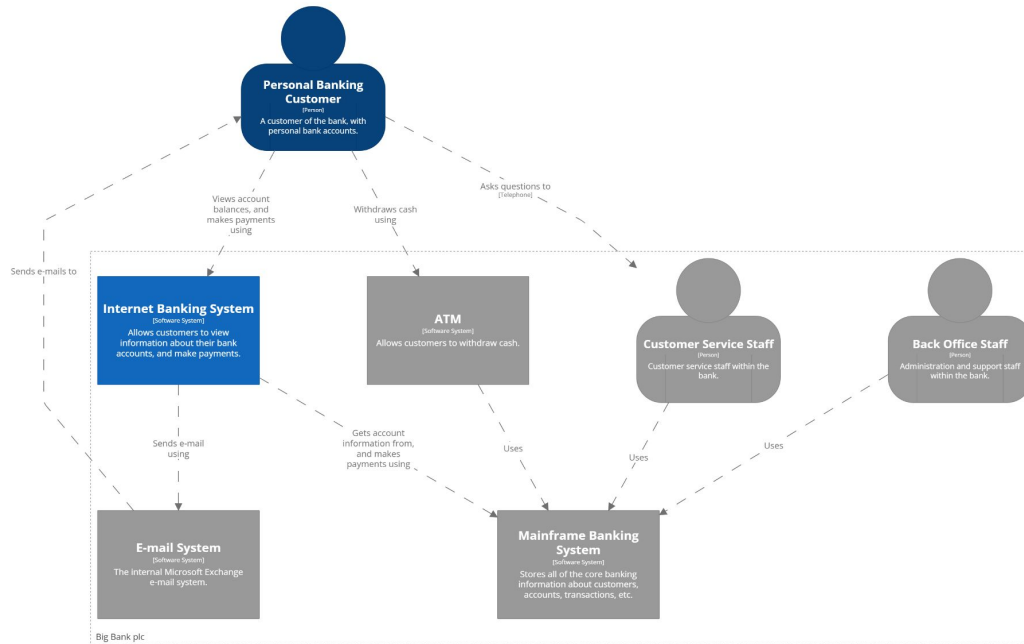
3	To do: Understand architecture “bugs”	<p>Эскиз архитектуры “Интернет-магазина” имеет ряд недостатков:</p> <ol style="list-style-type: none">1. Сервер API, моб. приложения вынесены в отдельные системы (здесь для наглядности), но должны быть контейнерами2. Таблицы и представления БД вынесены на уровень компонентов, но должны быть на уровне кода (здесь сделано для наглядности и компактности представления без уровня C4)3. По идее авторов модели C4, резервные системы не должны отображаться на диаграммах C1-C4, для этого предназначена <i>диаграмма развертывания</i>, о которой речь ниже
---	--	--

Вводный урок по Structurizr. Диаграмма ландшафта (4)

2

Open workspace.dsl and analyze the (“C0”) diagram

Откройте <https://structurizr.com/dsl?example=big-bank-plc> или скопируйте из https://dmpsy.club/references/Structurizr/009_DeploymentDiagram/workspace.dsl в Demo Page (<https://structurizr.com/dsl>).
Выведите и изучите диаграмму ландшафта банковской системы:



Вводный урок по Structurizr. Диаграмма развертывания

0	To do: Understand the deployment diagram concept and usage	<p>Диаграммы развертывания (deployment) конкретизируют варианты установки системы ПО (для разработки, тестирования, эксплуатации), основанные на UML Deployment Diagram (https://en.wikipedia.org/wiki/Deployment_diagram)</p> <p>Адресованы архитекторам ПО, разработчикам, архитекторам инфраструктуры и персоналу тестирования, эксплуатации и тех. поддержки</p>
---	---	--

Вводный урок по Structurizr. Диаграмма развертывания (2)

1	To do: Understand the deployment diagram structure	<p>Среда развертывания (deploymentEnvironment) помещается в конце области действия модели и состоит из одного (deploymentNode) или нескольких, возможно вложенных, узлов</p> <p>Узел развертывания представляет собой место, где запущен экземпляр (instance) программной системы / контейнера, то есть физическая (сервер или устройство), виртуализированная (IaaS, PaaS, виртуальная машина), или контейнеризированная инфраструктура (контейнер Docker), среда выполнения (например, сервер БД, веб-сервер/сервер приложений Java EE, Microsoft IIS) и т. Д.</p> <p>Среда обычно включает следующие элементы: deploymentNode, softwareSystemInstance, containerInstance, infrastructureNode</p>
---	---	--

Вводный урок по Structurizr. Диаграмма развертывания (3)

2

To do:
Analyze the
Development
deployment
environment
example
(пример
среды
разработки)

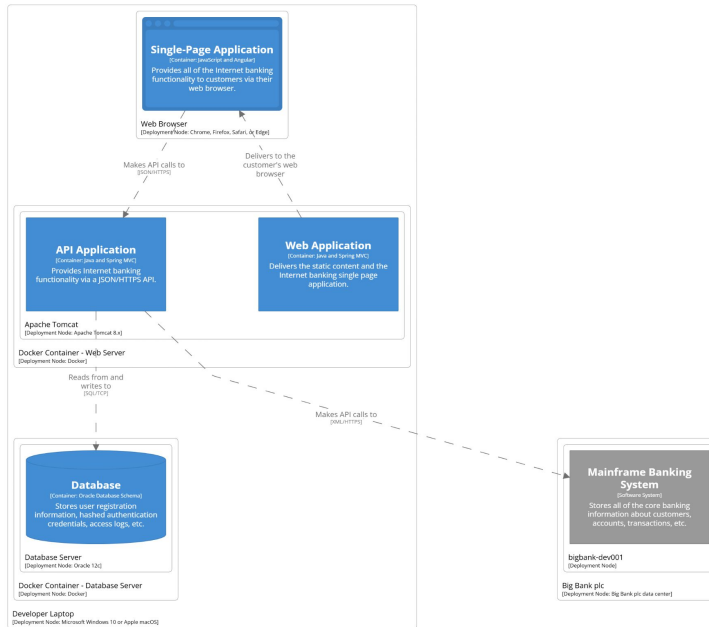
```
deploymentEnvironment "Development" {  
    deploymentNode "Developer Laptop" "" "Microsoft Windows 10 or Apple macOS" {  
        deploymentNode "Web Browser" "" "Chrome, Firefox, Safari, or Edge" {  
            developerSinglePageApplicationInstance = containerInstance singlePageApplication  
        }  
        deploymentNode "Docker Container - Web Server" "" "Docker" {  
            deploymentNode "Apache Tomcat" "" "Apache Tomcat 8.x" {  
                developerWebApplicationInstance = containerInstance webApplication  
                developerApiApplicationInstance = containerInstance apiApplication  
            }  
        }  
        deploymentNode "Docker Container - Database Server" "" "Docker" {  
            deploymentNode "Database Server" "" "Oracle 12c" {  
                developerDatabaseInstance = containerInstance database  
            }  
        }  
    }  
    deploymentNode "Big Bank plc" "" "Big Bank plc data center" "" {  
        deploymentNode "bigbank-dev001" "" "" "" {  
            softwareSystemInstance mainframe  
        }  
    }  
}
```

Вводный урок по Structurizr. Диаграмма развертывания (4)

3

Open workspace.dsl and analyze the deployment diagram

Откройте <https://structurizr.com/dsl?example=big-bank-plc> или скопируйте из https://dmpsy.club/references/Structurizr/009_DeploymentDiagram/workspace.dsl в Demo Page (<https://structurizr.com/dsl>).
Выведите и изучите диаграмму развертывания банковской системы:

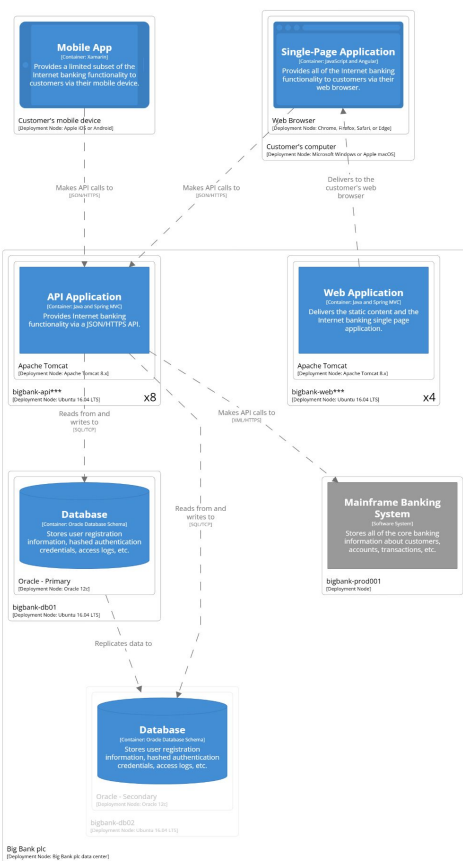


Развернута минимальная среда для разработки: внешний узел для mainframe datacenter и узел - PC разработчика с вебсервером для запуска single-page application и два docker-контейнера для API и Web-application на веб-сервере и контейнера БД

Особенности:

- Отношения между элементами построены автоматически
- Могут быть выведены три диаграммы, отражающие сценарий развертывания (animation) конфигурации

Вводный урок по Structurizr. Диаграмма развертывания (5)



4. Откройте <https://structurizr.com/dsl?example=big-bank-plc> или скопируйте из https://dmpsy.club/references/Structurizr/009_DeploymentDiagram/workspace.dsl в Demo Page (<https://structurizr.com/dsl/>).

Выведите и изучите диаграмму развертывания банковской системы:

Развернута полнофункциональная целевая конфигурация “продакшн”

- 1 mobileApplication установлено на телефоне (iOS / Android) клиента
- 2 singlePageApplication доступно в браузере на PC клиента
- 3 в datacenter Банка

- на двух веб-серверах Apache / Ubuntu запущены webApplication (серверная часть и статика) и apiApplication
- на двух БД Oracle / Ubuntu развернуты основной и резервный инстансы - схемы БД (учетные данные и логи пользователей) с репликацией данных в режиме реального времени
- развернуто центральное хранилище данных - mainframe (хранит всю основную банковскую информацию о клиентах, счетах, транзакциях и т. д.)

Особенности:

- Отношения между элементами построены автоматически
- Могут быть выведены диаграммы, отражающие сценарий развертывания (animation) конфигурации
- Резервирование БД определено отношением:

primaryDatabaseServer -> secondaryDatabaseServer "Replicates data to"

Вводный урок по Structurizr. Динамическая диаграмма

0	To do: Understand the dynamic diagram concept and usage	<p>Динамическая диаграмма отражает последовательность действий, сценарий реализации той или иной бизнес-функции. Она основана на UML Communication (ранее Collaboration) Diagram (https://en.wikipedia.org/wiki/Communication_diagram).</p> <p>Она похожа на UML Sequence diagram (https://en.wikipedia.org/wiki/Sequence_diagram), но допускает свободное расположение элементов диаграммы с пронумерованной последовательностью их взаимодействий.</p> <p>Адресованы широкому кругу специалистов, как техническим, входящим в команду разработки, так и бизнесу</p>
---	--	--

Вводный урок по Structurizr. Динамическая диаграмма (2)

1	To do: Understand the dynamic diagram structure	<p>Динамическая диаграмма может быть выведена либо в стиле Collaboration, либо Sequence-диаграммы. Оба представления равноправны и могут быть применены по Вашему желанию</p> <p>Диаграмма может включать системы ПО, контейнеры и компоненты, взаимодействующие между собой</p> <p>NB Динамические диаграммы следует использовать не повсеместно, а лишь для демонстрации специфических или повторяющихся сценариев или сложной логики взаимодействия элементов.</p>
---	--	--

Вводный урок по Structurizr. Динамическая диаграмма (3)

2

**To do:
Analyze the
dynamic
diagram
example
(пример
динамическо
й
диаграммы)**

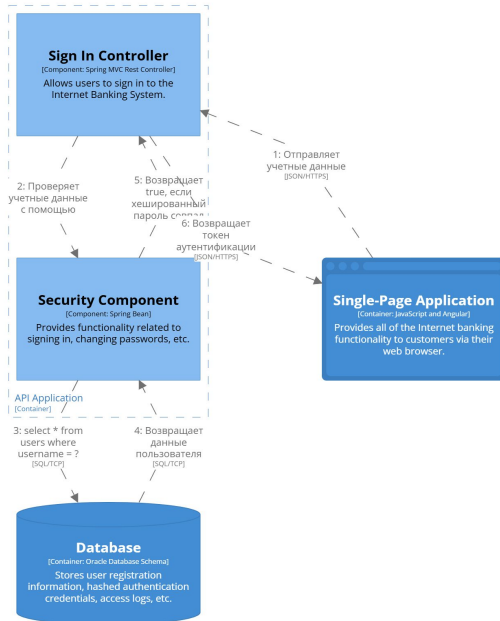
```
dynamic apiApplication "SignIn" "Сценарий функции входа пользователя в приложение" {  
    singlePageApplication -> signinController "Отправляет учетные данные"  
    signinController -> securityComponent "Проверяет учетные данные с помощью"  
    securityComponent -> database "select * from users where username = ?"  
    database -> securityComponent "Возвращает данные пользователя"  
    securityComponent -> signinController "Возвращает true, если хешированный пароль совпал"  
    signinController -> singlePageApplication "Возвращает токен аутентификации"  
    autoLayout  
    description "Сценарий функции входа пользователя в приложение"  
}
```

Вводный урок по Structurizr. Динамическая диаграмма (4)

3

Open workspace.dsl and analyze the dynamic diagram

Откройте <https://structurizr.com/dsl?example=big-bank-plc> или скопируйте из https://dmpsy.club/references/Structurizr/009_DeploymentDiagram/workspace.dsl в Demo Page (<https://structurizr.com/dsl>).
Выведите и изучите динамическую диаграмму входа пользователя:



1. Учетные данные пользователя пересылаются SinglePage Application в Sign In Controller
2. Sign In Controller отправляет их на проверку в Security Component
3. Security Component выполняет запрос в БД
4. БД возвращает результат запроса
5. Если пароль совпал, Security Component отправляет true в Sign In Controller
6. Sign In Controller предоставляет аутентификационный токен в Single Page Application

Особенности

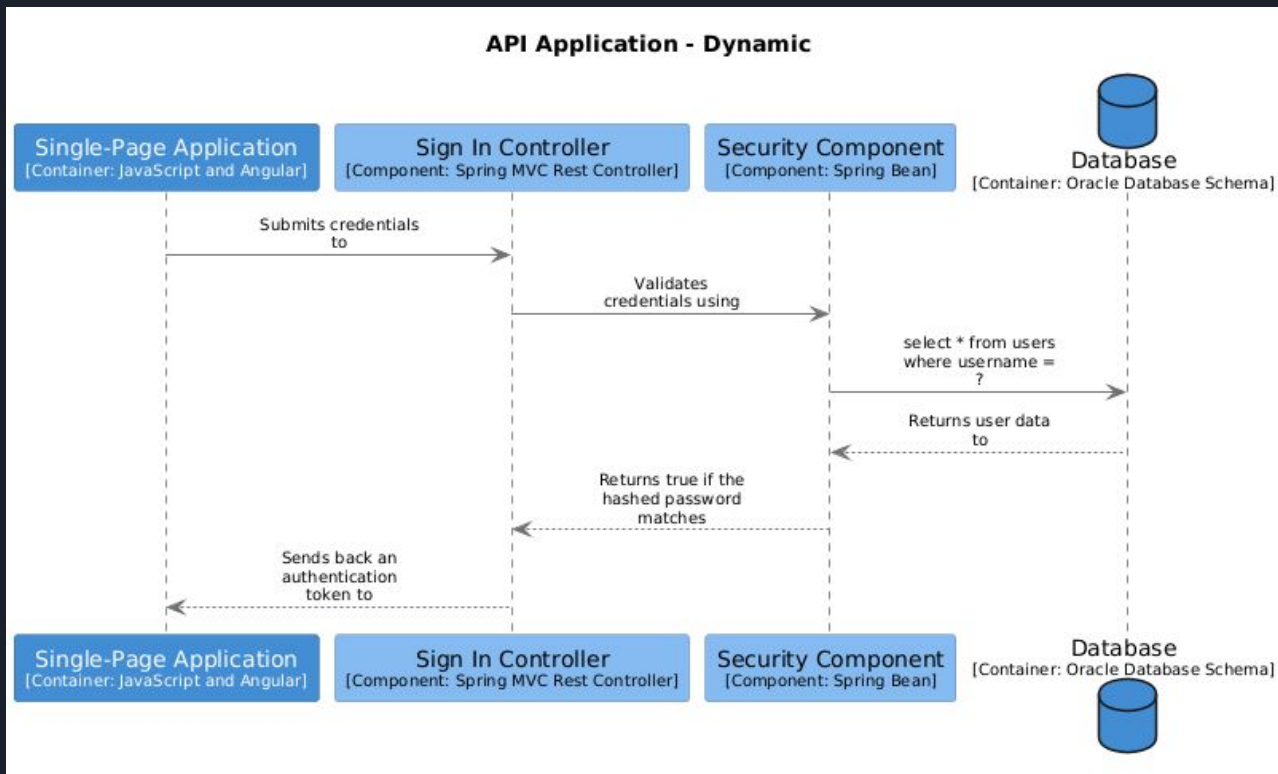
- Диаграмма выполнена в “collaboration style”, возможная альтернатива - “sequence style”, аналогичный представлению UML sequence diagram (на след слайде)
- Также возможен экспорт и отрисовка в PlantUML

Вводный урок по Structurizr. Динамическая диаграмма (5)

3

Open and analyze the (sequence style) dynamic diagram

Откройте <https://c4model.com/images/bigbankplc-SignIn-sequence.png> и изучите динамическую диаграмму входа пользователя (sequence style)



Подведем итоги

1	Обсудили базовые принципы иерархической модели C4
2	Проработали основные и вспомогательные диаграммы Structurizr в нотации DSL

Уровень	Диаграмма	Ключевое слово	Комментарий
“C0”	Landscape	softwareSystem	Ландшафт - окружение из (корпоративных) Систем ПО
C1	Context	softwareSystem	Разрабатываемая, тестируемая, поддерживаемая и внедряемая нами система ПО
C2	Container	container	Составная часть системы ПО, приложение или хранилище данных, развертываемая отдельно
C3	Component	component	Составная часть контейнера, работающая внутри него и НЕ развертываемая отдельно
C4	Code	image <component_name>	Классы, интерфейсы, объекты, функции того или иного конкретного языка программирования
	Deployment	deploymentEnvironment	Вариант установки системы ПО (для разработки, тестирования, эксплуатации)
	Dynamic	dynamic	Сценарий взаимодействия элементов системы ПО


Полезные ссылки / Useful Links

1	https://c4model.com/	Домашняя страница Модели C4
2	https://structurizr.com/	Домашняя страница Structurizr
3	https://docs.structurizr.com/	Документация Structurizr
4	https://structurizr.com/dsl	Демо-страница Structurizr
5	https://docs.structurizr.com/dsl/language#infrastructurenode	Словарь языка Structurizr DSL
6	https://www.youtube.com/watch?v=mqoU2C-USP0	Презентация Саймона Брауна (https://www.infoq.com/profile/Simon-Brown/#allActivity)
7	https://github.com/structurizr/	Structurizr в Github



Темы для Части 3 (сделаю по запросу)

1	Варианты установки Structurizr (Structurizr-lite, установка в Docker и т.д.)
2	Разбор стандартных примеров (архитектура микросервисов, интернет-магазин и т.д.)
3	Модель С4 - где применять, преимущества и ограничения, FAQ



Следующая лекция - (курс NextJS!) 8. Static and Dynamic Rendering / Статическая и динамическая отрисовка

Презентация доступна для скачивания здесь:

<https://dmpsy.club/references/Structurizr/StructurizrIntro.pdf>

Загрузить примеры из лекции:

<https://dmpsy.club/references/Structurizr/LectureExamples.zip>

Поддержать автора: <https://www.donationalerts.com/r/dmitrymak>



<https://dmpsy.club>

postmaster@dmpsy.club

<https://www.donationalerts.com/r/dmitrymak>

