

Recap



```
1 // SPDX-License-Identifier: MIT
2 pragma solidity ^0.8.0;
3 contract A{
4     uint x;
5     function setX(uint _x) public { x=_x;}
6     function getX() public returns (uint){ return x;}
7 }
8 contract B{
9     function useNew() public returns (uint) {
10         A a = new A();
11         a.setX(100);
12         return a.getX();
13     }
14 }
```

smerf.net

CST4125:L17

Winter 2023

6 / 48

Object-Oriented Programming



- Contract \equiv class
- Implementation of a contract is an object
- Behaviour
- Structure
- UML: Class diagrams become Contract Diagrams
- Contracts can inherit
- Revision for UML, see [3]

smerf.net

CST4125:L17

Winter 2023

7 / 48

Constructor



- What address is used, when creating a new instance?
- Constructors, are default
- Declare constructors
- Only one constructor, no overloading
- Constructor has the same name as the contract
- Constructor cannot return any values
- Can set state variables to default values

smerf.net

CST4125:L17

Winter 2023

8 / 48

Contracts Diagram

taken from [1]



Contract Name
+ State Variables
+ Functions

SimpleCoin
+ coinBalance: mapping
+ allowance: mapping
+ frozenAcct: mapping
+ owner: address
+ transfer(address, uint): void
+ authorise(address, uint): void
+ transferFrom(address, address, uint): bool
+ mint(addressRecipient, uint): void
+ freezeAcct(address, bool): void

smerf.net

CST4125:L17

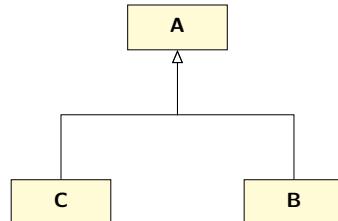
Winter 2023

9 / 48

Contracts and Single Inheritance



```
1 contract A{}
2
3 contract B is A{}
4
5 contract C is A{}
```



smerf.net

CST4125:L17

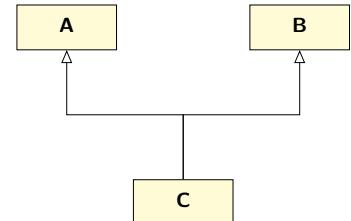
Winter 2023

10 / 48

Contracts and Multiple Inheritance



```
1 contract A{}
2
3 contract C is A,B{}
```



smerf.net

CST4125:L17

Winter 2023

11 / 48

Interface



- Similar to Java
- Defines all methods for a contract to implement
- Structure and consistency
- Interface does not implement any methods
- Can define complex types, not state variables
- Can't define variables
- Can't define constructors
- Can't define modifiers
- Code reuse
- Can inherit other interfaces, not other Contracts

send



call



Address Functions



Address

- address.balance
- address.transfer(amount)
- address.send(amount)
- address.call(payload)
- address.callcode(payload)
- address.delegatecall()

transfer



Fallback



Definition

Something to which one can resort or retreat

- Using a smart contract's functions
- Make a call to a function that does not exist?
- When the fallback function is invoked

Fallback



Definition

Something to which one can resort or retreat

- Using a smart contract's functions
- Make a call to a function that does not exist?
- When the fallback function is invoked

Solidity Definition

A fallback function is invoked when the function call does not match any of the function names

Application Development Lifecycle Management



- Design, Develop, Deploy & Test
- Requirements capture
- Architects documents these
- Composed of sprints
- Eventually, coded into contracts, data and functions
- Deployed
- Tested
- Need for tools

Installing Truffle



- VM has Truffle pre-installed
- Truffle
- Debian:
 - sudo apt update
 - sudo apt install nodejs
 - sudo apt install npm
 - sudo npm install -g truffle
 - truffle --version

Fallback Rules

```
1 // SPDX-License-Identifier:  
    MIT  
2 pragma solidity ^0.8.0;  
3 contract fallbackExample{  
4     function () { uint a  
        =10; }  
5 }
```

- Only one fallback function per contract
- No parameters
- No function name
- No return values
- Gas required to execute a fallback function
- Limit Gas consumption to 2,300Wei
- Test fallback function does not exceed gas consumption
- Security lapses
- Can have a payable fallback function

Truffle



- Application Lifecycle Management (ALM)
- Need supporting tools
- Decentralisation is different
- Truffle accommodates some of these differences
- Help in the DevOps process
- Truffle is a utility for blockchain development that:
 - Works with Ethereum and other blockchain development environments
 - Compile smart contracts
 - Deploy smart contracts
 - Test smart contracts
- Overall: makes production more efficient

Ganache



- Another Tool
- Decentralisation
- Accounts, Transactions and Blocks
- User End
- GUI and CLI
- Installation (Debian) & Implementation:
 - Visit Ganache
 - Download and move to a new folder in Home directory, Ganache
 - Debian systems not supporting FUSE, sudo apt install libfuse2
 - Double click AppImage file

Ganache

Quickstart

CREATE A WORKSPACE
Quickstart for a one-click blockchain or create a new workspace for advanced setup options.

QUICKSTART **NEW WORKSPACE**

smerf.net CST4125:L17 Winter 2023 29 / 48

Ganache

Accounts

ADDRESS	BALANCE	DECIMALS	TX COUNT	INFO
0xA63222fe7994e94d8CBeE90c655f362e488eEE8f	100.00 ETH	0	0	View
0x1f55a76C8f600b1230cFF383a6d62d6d11C877C10	100.00 ETH	0	1	View
0xb099c158c9f4fb730bbab0C<0E5Ab10B7e35F00	100.00 ETH	0	0	View
0x54261CC6eb139316EfafC00831Aecad1405E272	100.00 ETH	0	3	View
0x7f0E7999559898BD2e16012871f506c0852469	100.00 ETH	0	4	View
0xC9Ef4E7438DABEB176986130FB4D4A36191deE8	100.00 ETH	0	5	View
0xE0FB446E20E17aEa72728F74713D43F7cD725B4C	100.00 ETH	0	6	View
0x9Ef7F9b5c94Ed142fd85064e495026c7E5f0f0E27	100.00 ETH	0	7	View
0x8sd289f8dA197axd2e1F58d98846499Af93C868	100.00 ETH	0	8	View

smerf.net CST4125:L17 Winter 2023 30 / 48

Ganache

Blocks

BLOCK	NUMBER	TIME	TRANSACTIONS
0	00	2022-06-28 09:51:31	0

ALL TRANSACTIONS

smerf.net CST4125:L17 Winter 2023 31 / 48

Ganache

Transactions

NO TRANSACTIONS

smerf.net CST4125:L17 Winter 2023 32 / 48

Ganache

Contracts

NO PROJECTS IN WORKSPACE

Unlock more information about your smart contracts by linking a Truffle project. View deployed contract addresses, associated transactions, decoded events, and even contract state. Your contract data will update in real time during development.

[LINK TRUFFLE PROJECT](#)

smerf.net CST4125:L17 Winter 2023 33 / 48

Ganache

Events

NO EVENTS

smerf.net CST4125:L17 Winter 2023 34 / 48

Ganache Logs



smerf.net CST4125:L17 Winter 2023 35 / 48

Ganache and Remix

Initialise & Setup



Ganache and Remix

Initialise & Setup

The screenshot displays the Geth GUI interface, specifically the Contracts tab. It shows the deployment of a Solidity smart contract named 'Storage' with the following code:

```

1 // SPDX-License-Identifier: MIT
2 pragma solidity ^0.8.0;
3
4 contract Storage {
5     uint256 number;
6
7     function store(uint256 newValue) public {
8         number = newValue;
9     }
10
11     function retrieve() public view returns (uint256) {
12         return number;
13     }
14 }

```

The contract has been deployed at address 0x832f7e7094e04bcb8e90dc0e153f362d4+ and has a balance of 100.00 Ether. The current value stored is 0.

smerf.net CS14125:L17 Winter 2023 37 / 48

Ganache and Remix Transactions



Ganache

The screenshot shows the Remix Ethereum IDE interface. On the left, the code editor contains the following Solidity code:

```

// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;

contract Storage {
    uint256 number;

    function store(uint256 num) public {
        number = num;
    }

    function retrieve() public view returns (uint256) {
        return number;
    }
}

```

The 'DEPLOY & RUN TRANSACTIONS' tab is active. In the 'ENVIRONMENT' dropdown, 'Ganache Provider' is selected. The 'DEPLOYMENT' dropdown shows 'DIALES.JEET/50740460'. The 'GAS LIMIT' is set to 3000000. The 'Value' field is empty. The 'Contract' dropdown shows 'Storage - contractId: Storage 0'. The 'Deploy' button is highlighted in orange. Below the code editor, the 'Transactions recorded' section is shown.

On the right, the Ganache interface shows a transaction history table:

ID	From	To	Gas used	Value
0	0x0d84780f265f3b43e190516c405f157ea501a8e2f25d65eaef23fb132370b79c	0x00	8	0

At the bottom, the status bar indicates '20 Jun 10:33' and the footer says 'Remix IDE v0.9.10 (forked)'.

Digitized by srujanika@gmail.com

Remix Desktop



- Offline resource
- Link up to Ganache
- Have all functionality of remix online
- Debian Installation:
 - Remix-ide
 - Download AppImage (note about FUSE on previous slides)
 - Create a remix sub-directory in the home directory
 - Move the AppImage file to the remix directory
 - Change the permissions of the file to allow execution (`chmod fileName.AppImage`)
 - Double click (and be patient and wait)

A set of small, light-blue navigation icons typically found in presentation software like Beamer. They include symbols for back, forward, search, and table of contents.

Ganache and Remix Blocks



Ganache and Remix Blocks

The screenshot shows two separate browser windows side-by-side, both connected to the same Hyperledger Fabric instance.

Left Window (Deploy & Run Transactions):

- Environment:** Ethereum Provider
- Deploy:** Deployed
- Access:** Accessible
- Address:** `HTTP://192.168.0.10:7050`
- CallSite:** `0x00`
- Contract:** Storage
- Storage - contract1.Storage.ind:** `Storage`
- Deploy:** Deploy
- Push to BPS:** Off
- In Address:** `0x00`
- Transactions received:** 0
- Deployed Contracts:** STORAGE@00
- Block:** `[Block 1] [Storage]` from `0x00` at `0x00`
- Last transaction:** `0x00`
- Block:** `[Block 2] [Storage]` from `0x00` at `0x00`

Right Window (Blocks):

Block	Height	Timestamp	Hash	Previous Hash	Block Size	Block Type	Block Status
1	1	2022-06-28 11:18:12	5413009	1256717	1040	DATA	VALID
2	2	2022-06-28 11:18:12	5413010	5413009	1040	DATA	VALID

Winter 2023 38 / 48

Send Ether



- Contract with 1 address: EOA
- Send ether from EOA to a nominated address
- Include a value of 1 ether to transfer

Send Ether

```

1 // SPDX-License-Identifier: GPL-3.0
2
3 pragma solidity >=0.7.0 <0.9.0;
4
5 contract sendEther{
6
7     function sendViaTransfer(
8         address payable _to) public payable{
9         _to.transfer(msg.value)
10    }
11 }
```

smerf.net

CST4125:L17

Winter 2023 41 / 48

- Contract with 1 address: EOA
- Send ether from EOA to a nominated address
- Include a value of 1 ether to transfer



Send Ether Example

From Account	To Account	Value (ETH)	Index
0x4025f276cf860b1230cff303e6d2d6011	0xb099c1582c9f4fb736bb0c4c0e54b188	100.00	0
0xb099c1582c9f4fb736bb0c4c0e54b188	0x54261cc6ab1933161faFcD831Aeacaed1	100.00	1
0xb099c1582c9f4fb736bb0c4c0e54b188	0x8f87f909b59998b102e16b12071fb06cd	100.00	2
0xb099c1582c9f4fb736bb0c4c0e54b188	0x91de68	100.00	3
0xb099c1582c9f4fb736bb0c4c0e54b188	0x6f8b46e20e17ae72728f74713d43f7c	100.00	4
0xb099c1582c9f4fb736bb0c4c0e54b188	0x72394c	100.00	5

smerf.net

CST4125:L17

Winter 2023 42 / 48

Send Ether Blocks



TX HASH	FROM ADDRESS	TO CONTRACT ADDRESS	VALUE (WEI)
0x50df76fe6a7a2a2b20fb544c26f00bccb8ff9fb2d8a2d1f249f43979daef09a7	0x1958a8e7578ecc5a3c811488853627d73fe072d9d128dc82b113e8d	0x00	2000

smerf.net

CST4125:L17

Winter 2023 43 / 48

Send Ether Transactions

TX HASH	FROM ADDRESS	TO CONTRACT ADDRESS	VALUE (WEI)
0x50df76fe6a7a2a2b20fb544c26f00bccb8ff9fb2d8a2d1f249f43979daef09a7	0x1958a8e7578ecc5a3c811488853627d73fe072d9d128dc82b113e8d	0x00	2000
0xb730e2aa3234f4f71542fd17792c4a3105a4ead7979916798e8ff91d99832e	0x4432171f994a80c860515732a08484cf	0x00	2000
0xb730e2aa3234f4f71542fd17792c4a3105a4ead7979916798e8ff91d99832e	0x4432171f994a80c860515732a08484cf	0x00	2000
0xb4998db0fce087269833677887887081aa88fc0ead5431dc7928b4d33c8d51	0x4432171f994a80c860515732a08484cf	0x00	2000
0xb4998db0fce087269833677887887081aa88fc0ead5431dc7928b4d33c8d51	0x4432171f994a80c860515732a08484cf	0x00	2000
0xb25a550bae2714e42a461c6d277889f693b01632bdc9a31dc73ac7e51e	0x4432171f994a80c860515732a08484cf	0x00	2000
0xb25a550bae2714e42a461c6d277889f693b01632bdc9a31dc73ac7e51e	0x4432171f994a80c860515732a08484cf	0x00	2000
0xb6c24c8acfda2b0dd6fe0b95862c0b00233a07accf55621ea779f6dc0bf4f1	0x4432171f994a80c860515732a08484cf	0x00	2000

smerf.net

CST4125:L17

Winter 2023 44 / 48

Send Ether Transaction Details



TX HASH	FROM ADDRESS	TO CONTRACT ADDRESS	VALUE (ETH)
0x050df76fe6a7a2b20fb544c26f00bccb8ff9fb2d8a2d1f249f43979daef09a7	0x1958a8e7578ecc5a3c811488853627d73fe072d9d128dc82b113e8d	0x00	1.00

smerf.net

CST4125:L17

Winter 2023 45 / 48

Summary

- Private Networks
- Ganache
- Truffle
- Remix-ide (Desktop)
- Contract as a class (OOP)
- Fallback
- Transfer



Winter 2023 46 / 48

Reading



Read Chapter 6 & 7 in [2]

References I

- [1] R. Infante. *Building Ethereum Dapps*. Manning, 2019. ISBN: 9781617295157.
- [2] Ritesh Modi. *Solidity Programming Essentials*. Packt, 2018. ISBN: 978-1-78883-138-3.
- [3] Martina Seidl et al. *UML@ classroom: An introduction to object-oriented modeling*. Springer, 2015.