

# Virtuální herní světy

## Svět s hráčem a bez něj

Radek Richtr

October 19, 2020

*sluha: Veličenstvo, něco se děje!*

*král Pikola: Něco se děje... To je normální. Jinak by nebyly dějiny.*

# Outline I

## Dělení

- ① Timeless
- ② Causal
- ③ Induced
- ④ Induced
- ⑤ Generovaný svět  
Gramatiky
- ⑥ Evolving world  
E.I.W  
Evolving Independent World
- ⑦ (Open) World

# Vývoj

Herní světy jako prostředí pro hry se značně (až prudce) vyvíjí.

Hlavními omezeními byly dříve především technické možnosti, ale zřejmou roli hrál i mindset tvůrců, především v definici vztahu *hráč* - *svět*.

Zmiňme, že jako většina přístupů je i toto dělení z principu nepřesné, existují výjimky a hry vyskytující se na pomezí zde vydělených základních oblastí.

# Vývoj

## Prehistorie (?)

- Timeless
- Causal

## Historie (?)

- Player Induced

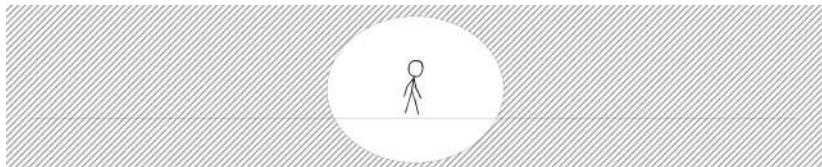
## Současný stav

- Evolving (Independent) World

# Timeless

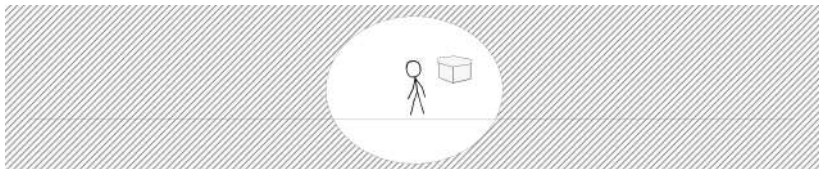
Nejjednodušší přístup ke světu je ten, kdy svět v zásadě existuje *jen* v té oblasti, kterou je avatar (potažmo hráč) je schopen vnímat.

- dohled
- obrazovka
- level





# Timeless



## Bezčasý svět

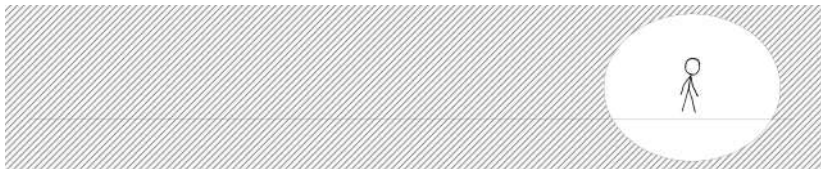
Nezáleží na tom, kdy do které části světa hráč vstoupí, dokonce nezáleží na tom, co v které části hráč provedl (má-li to vůbec povoleno). Svět mimo vnímání avatara (hráče) se vrací do své bezčasé formy, ze kterého jej vytrhne až znovu avatarova přítomnost.

Je toto opravdu minulost? A jak se to projevuje dnes?





# Timeless

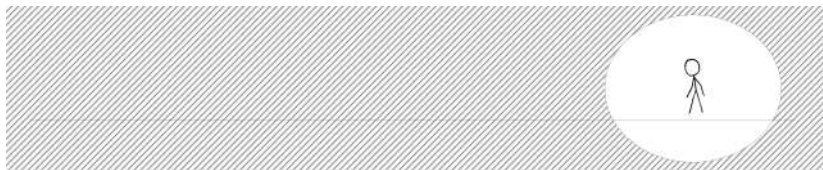


## Bezčasý svět

Nezáleží na tom, kdy do které části světa hráč vstoupí, dokonce nezáleží na tom, co v které části hráč provedl (má-li to vůbec povoleno). Svět mimo vnímání avatara (hráče) se vrací do své bezčasé formy, ze kterého jej vytrhne až znovu avatarova přítomnost.

Je toto opravdu minulost? A jak se to projevuje dnes?

# Timeless



## Bezčasý svět

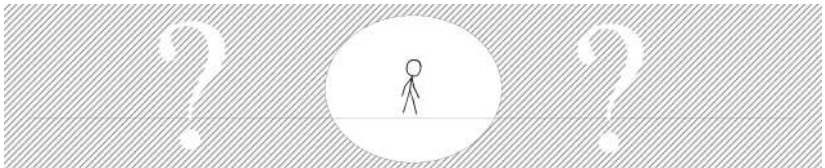
Nezáleží na tom, kdy do které části světa hráč vstoupí, dokonce nezáleží na tom, co v které části hráč provedl (má-li to vůbec povoleno). Svět mimo vnímání avatara (hráče) se vrací do své bezčasé formy, ze kterého jej vytrhne až znovu avatarova přítomnost.

Je toto opravdu minulost? A jak se to projevuje dnes?



# Timeless - generovaný

V extrémním případě je definice světa za hranicí vnímání avatara neznámá i samotné hře (není ani nikde v paměti) a nachází se tedy v neurčitém stavu.



## Timeless - generovaný

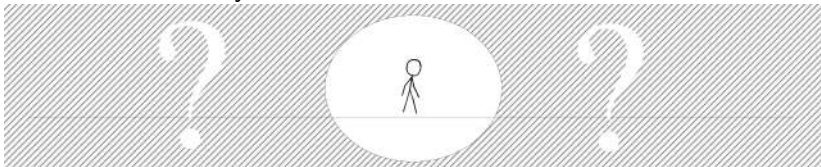
Obdobné jsou hry, kde je prostředí vytvořeno sadou pravidel [DB11] generativní gramatiky. Hra sice zná pravidla, ale je vždy potřeba vygenerovat celý level (jeho část).

Oblasti kterými prochází avatar jsou jen realizace funkcí generujících pseudonáhodné prostředí (vizte třeba DP Hájková) a po opuštění opět zapomenuty.

Neplést gramatiku s gramatikou!

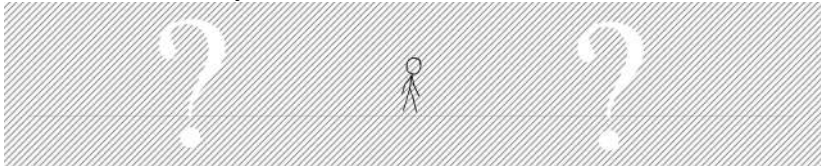
# Timeless - generovaný

Nahrání uložené hry:



# Timeless - generovaný

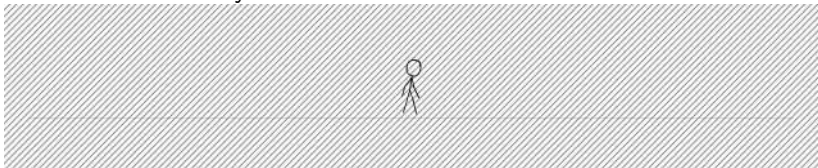
Nahrání uložené hry:



V čem je problém?

# Timeless

Nahrání uložené hry:



V čem je problém?



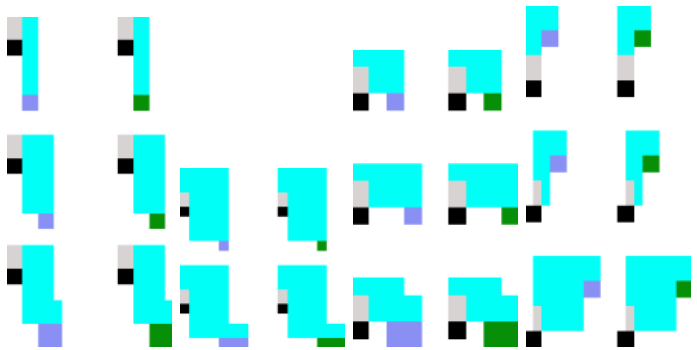
# Timeless - generovaný: Příklad



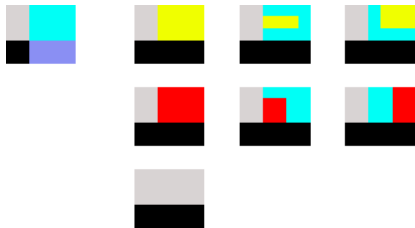
# Timeless - generovaný: Příklad



# Timeless - generovaný: Příklad














# Timeless - generovaný: Příklad



**Figure:** Grafické znázornění pravidel pro přidání chůze i s dalšími entitami. Vlevo: původní stav s plošinou a prázdným místem. Vpravo: možné další stavy pravidel. První řádek demonstruje přidávání bonusů, druhý řádek monster stojících na platformě, spodní řádek pak rozšíření stávající platformy.

# Timeless - generovaný: Příklad

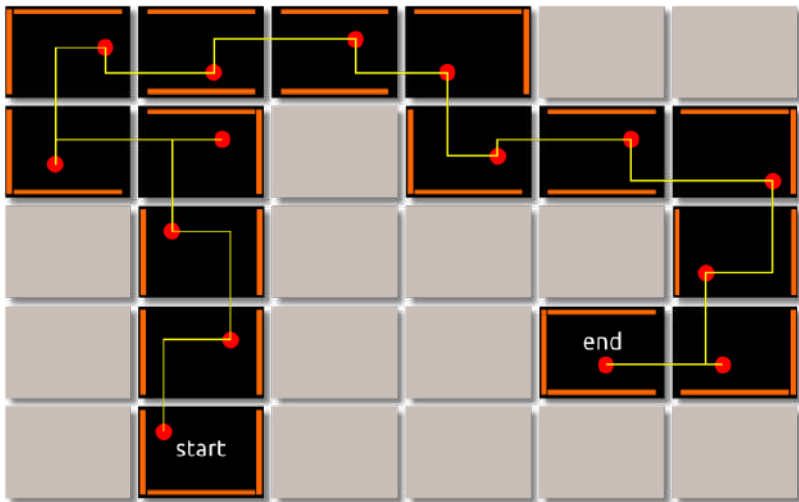
Table: Významy použitých barev

	terminál platformy
	cokoliv (pravidlo tato pole nezajímají)
	bonus
	monstrum
	neterminál nehmotné entity
	terminál průchozí části
	terminál platformy umístěný v aktuálním kroku do mřížky
	ilustrace neprůchozí části aktuální cesty
	ilustrace pevných entit (platform) aktuální cesty
	prolnutí průchozích částí aktuální cesty (růžová) a již existující cesty (šedá)
	aktuální požadavek přidání platformy
	teleport

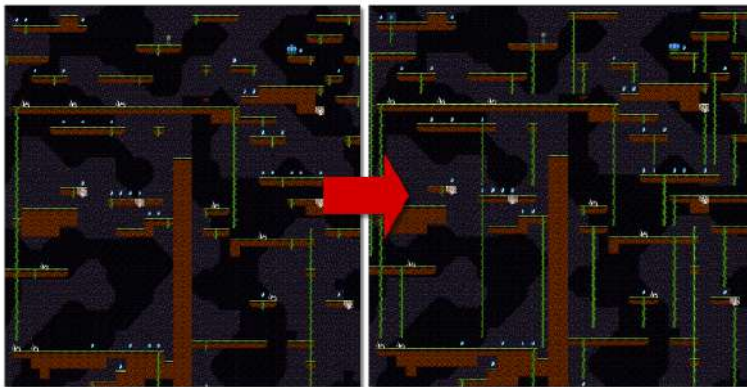
# Timeless - generovaný



# Timeless - generovaný



# Timeless - generovaný





# Contra

Důvodů pro vytvoření takového světa je mnoho, ty nejzákladnější omezení jsou (byla) paměťová. V dobách 8-bit, kdy celá hra zabrala maximálně desítky kB a existovala například omezení na maximální počet barev zobrazených najednou to jinak nešlo.



link: [Contra, 128kB včetně grafiky a hudby](#)

# G.I. Joe: A Real American Hero (NES)



link: Contra, 216kB včetně grafiky, textu, několika postav, animací, hudby

# Causal

Pro kauzální světy platí podobná pravidla jako výše, jen v tomto případě je část světa prošlá hráčem nastavena (zabití protivníků, otevřené dveře, atp.).

Svět je tedy popsán i několika proměnnými, kdyby se hráč potřeboval vrátit.

# Causal - generovaný

Hry s generovaným obsahem tedy svět vytváří (bud' iterativně, nebo dávkově) a přítomností hráče se svět dostane do určitého stavu.



# Causal - generovaný

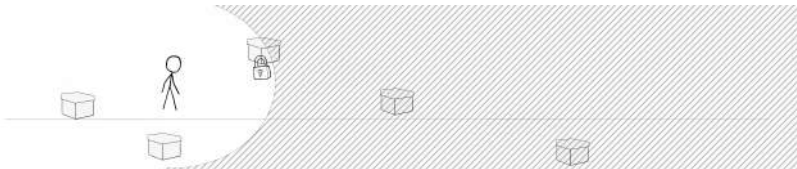
Kombinace vytváří zajímavý svět, podporující znovuhratelnost.  
Ikonickým příkladem je zde řada Diablo.



# Causal



# Causal

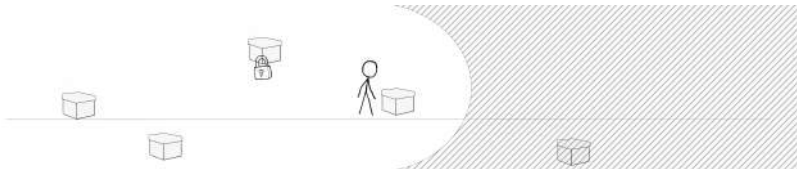


# Causal





# Causal



# Causal



# Causal



# Causal



# Causal



# Causal



# Player induced

S dostatkem paměti i prostředků přibyla možnost vytvářet komplexní světy a nechat je hráče více ovlivňovat.

Svět je v zásadě stále statický až na několik výjimek. Hráčovy akce mohou ovlivňovat i podobu světa který ještě nenavštívil.

Toto v kombinace "chytrým" level designem a dynamickým získáváním abilit umožní poměrně komplexní (opakované) procházení herního světa.

Hry se světem v tomto duchu působí obvykle heroicky (ve co se děje ovlivňuje jen hráč) ale zároveň i poměrně ploše.

# Player induced

Možnost ovlivňovat a brát svět jako celek vedl k poměrně divoké snaze jej takto využít. Opakovat použití levelů “jinak”, přidat nové možnosti dávající význam a nové využití již navštívených scén.

Někde tu může být velký problém, kde?



# Příklad



# Příklad



# Příklad



# Příklad



# Příklad



# Příklad



# Příklad



# Příklad





# Příklad



# Příklad



# Příklad



# Příklad



# Player induced - možný problém

## Svět

- se lehce mění
- reaguje na akce hráče
- je prochážitelný zpětně

# Player induced - možný problém

## Svět

- se lehce mění
- reaguje na akce hráče
- je prochážitelný zpětně

## Hráč

- získává nové předměty
- získává nové schopnosti (na ovlivňování světa)
- schopnosti se týkají i pohybu po světě



# Player induced - možný problém

## Vytváření párů stavů klíč - zámek

- Je obsah za zámkem nutný?
- Jak hráč zjistí, že potřebuje klíč?
  - Explicitní vysvětlení.
  - Utajení.
  - Tápání.



# Causal



# Causal



# Causal



# Causal



# Causal



# Causal



# Causal



# Causal





# Causal



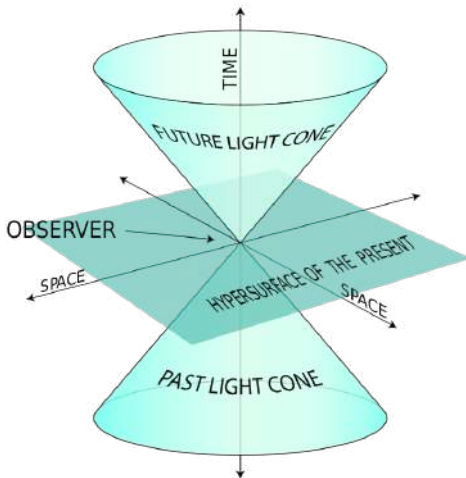
# Causal



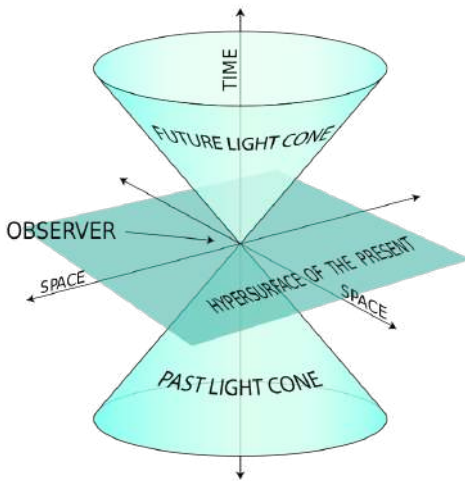
# Causal



# Úkrok stranou?



# Úkrok stranou



Nejaký vztah k VHS?

# PG vs. PCG

- Procedurální generování (PG) je dle [pcg19] proces, při kterém je procedurálními metodami tvořen obsah do hry, který však nijak zásadně neovlivňuje samotnou hratelnost. Např. textury, hudba, animace atd.
- Procedurální generování obsahu (PCG) je naopak dle [pcg19] generování obsahu, který přímo ovlivní hratelnost, tedy např. generování levelů (jejichž poskládání ovlivní samotný průběh hry), příběh, statistiky u zbraní, jejich síla atd. Někdy lze narazit i na zkratku PCG-G (Procedural Content Generation for Games) [HMVDVI13] označující stejnou oblast.

K Tématu lze doporučit buď zastřešující (a skvělou!) rešerší z práce Petry Svíčkové [Svi19], případně klíčové zdroje jako [Dou08, STN16, aMM11, for, HMVDVI13] atd.

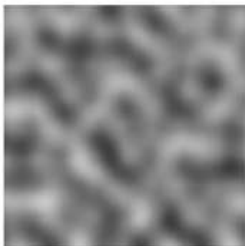
# P(C)G

Šest skupin metod procedurálního generování herního obsahu:

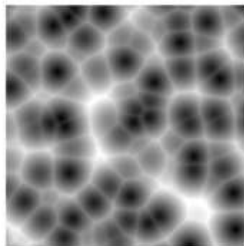
- pseudonáhodné generování čísel – např. *Perlinův šum*,
- generativní gramatiky – např. *L-systémy, dělící gramatiky, tvarové gramatiky*,
- filtrování obrazu – např. *binární morfologie a konvoluční filtry*,
- prostorové algoritmy – např. *dlaždicování a vrstvení, mřížkové dělení, fraktály, Voronoi diagramy*,
- modelování a simulace komplexních systémů – např. *celulární automat, tenzorové pole, simulace agenta a další*,
- umělá inteligence – např. *genetické algoritmy, umělá neuronová síť a plánování*.

# Perlinův šum

Asi nejčastěji používaná technika [Cep11] na generování (terénu). Alternativně jde použít třeba Worleyho [Joh10] (celulární) šum atd.



Perlin



Worley

**Figure:** Porovnání Perlinova a Worleyho šumu. [Cep11]



# Perlin



Figure: Použití Perlinova šumu pro generování 2D terénu [?].

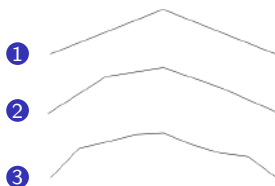


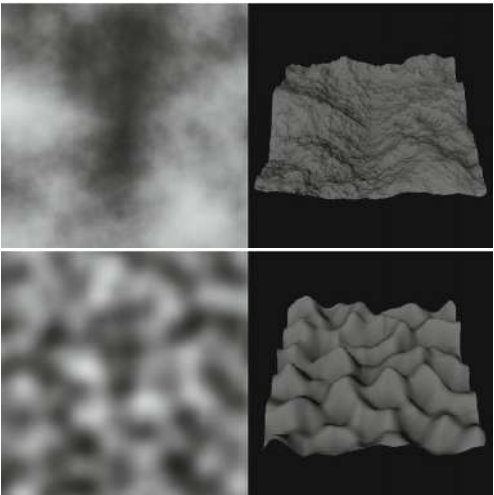
Figure: Postupné generování terénu algoritmem Midpoint Displacement [Mar].

# Perlin

Ikonickým případem aplikace 3D Perlinova šumu je Minecraft (terén, jeskyně).



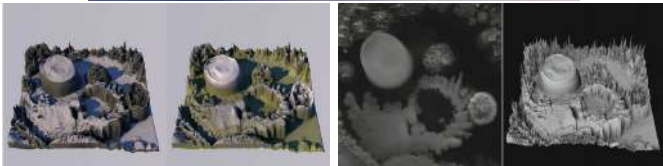
# 2D Perlin noise



# Perlin



# 2D Perlin noise



# Timeless - Gramatiky

- Level → Překážka + poklad
- Překážka → klíč + Překážka + zámek + Překážka
- Překážka → monstrum + Překážka
- Překážka → chodba
  
- k-m-c-z-m-z-p
- k-m-k-c-z-m-c-z-c-p
- c-p
- m-m-m-m-c-p

Moc krátké? Nezajímavě? Jak řešit?

# Timeless - Gramatiky

## Pravidla:

- Level → Překážka + poklad
- Překážka → klíč + Překážka + zámek + Překážka
- Překážka → monstrum + Překážka
- Překážka → chodba

## Možné realizace:

- k-m-c-z-m-z-p
- k-m-k-c-z-m-c-z-c-p
- c-p
- m-m-m-m-c-p

## Moc krátké? Nezajímavě? Jak řešit?

- Level → Překážka + Překážka + Překážka + poklad
- Level → strážce levelu + Překážka + mini boss + odměna + Překážka + klíč + Překážka + zámek + boss + poklad

# Timeless - Gramatiky

- Jednotlivá pravidla gramatiky mohou být vytvořeny ručně, nebo mohou být šlechtěna evolučními algoritmy.
- Vhodnou volbou mohou být L-systémy[Lin68, Moz09] generující poměrně zajímavé výsledky[ABG05]. Vhodné jsou nejen k modelování přírodních struktur, ale třeba i měst[PM01].
- Spíše než klasické řetězcové, nebo arborescentní gramatiky jsou vhodné grafové, nebo 2D gramatiky[A<sup>+</sup>02, RS95].

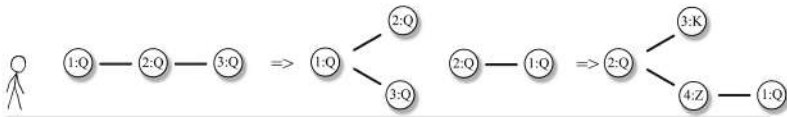


# Timeless - gramatiky



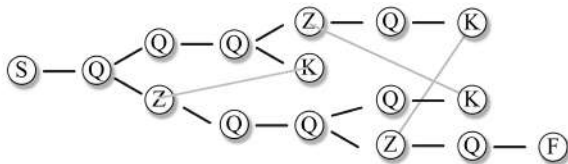
Legenda	
S	Start
Q	Místnost
Z	Zámek
F	Cíl
K	Klíč

# Gramatiky



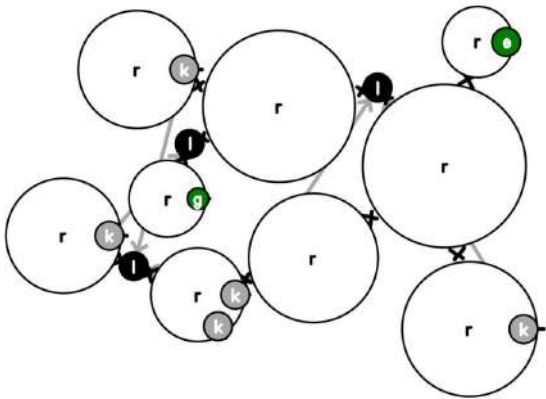
# Gramatiky

?!?





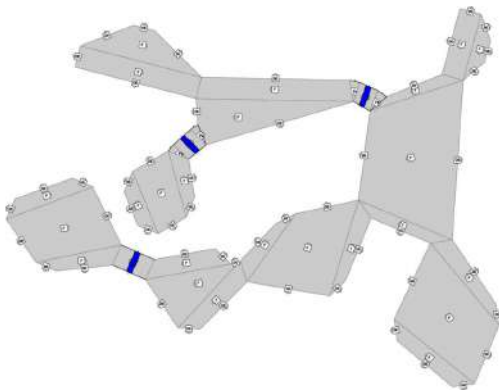
# Gramatiky



Vygenerování místností různé velikosti

Převod logického schématu na fyzický: Fyzikální prostor (objemová reprezentace)

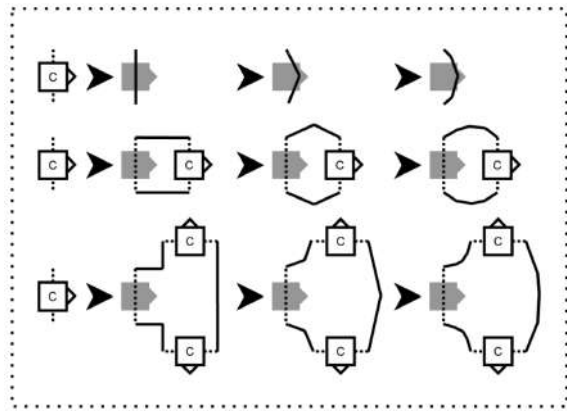
# Gramatiky



Obvodová reprezentace

Převod logického schématu na fyzický: Prostorová konstrukce místností a dveří (de facto přechod o obvodové reprezentaci)

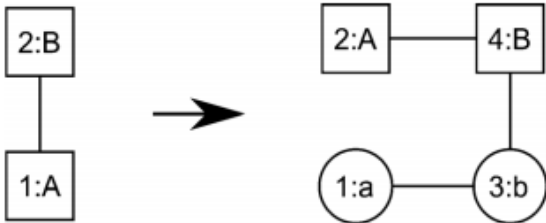
# Gramatiky



Obvodová reprezentace

Převod logického schématu na fyzický: Přesnější obvodová reprezentace

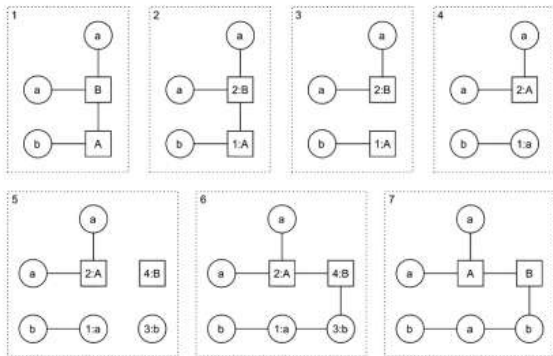
# Grafové gramatiky



Přepisovací pravidlo, kolečka jsou terminály, čtverce neterminály



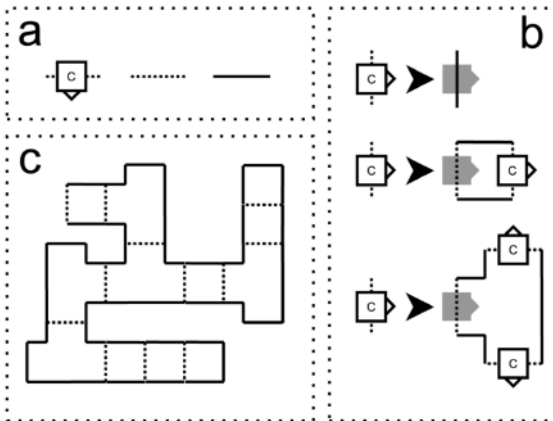
# Grafové gramatiky



Přepis pravidla, vygenerování konkrétního obsahu a/nebo propojení může být definované, nebo náhodné.

# Grafové gramatiky

## Triviální gramatika



# Příklady

- Elite (1984)
- SpeedTree (2001)
- Dwarf Fortress (2006)
- Terraria (2011)
- No man's sky (2016)
- Dead Cells (2018)

# Evolving world

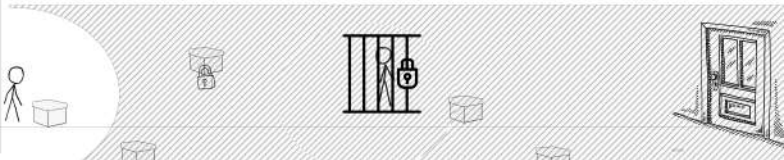
Vyvíjející se svět je ovlivňován nejen samotným hráčem, ale i časem.

S přidáním času jako faktoru je vytvořena zcela nová dimenze umožňující změnu.

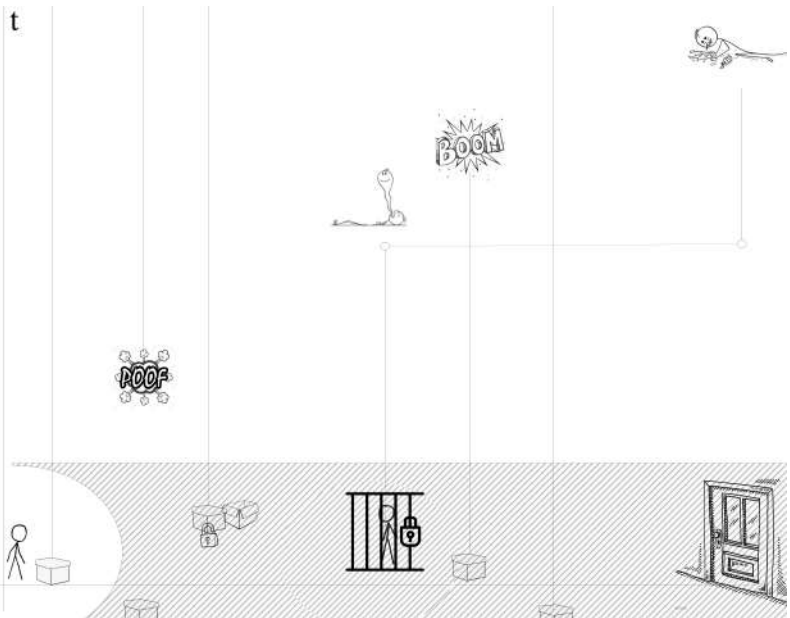
- NPC postavy se mohou dopouštět vlastních akcí
- změna množství surovin
- změna podoby světa
- změny ve světě nemusí být napořád
- atd.

# Příklad

t



# Příklad



# Evolving world

Vyvíjející se svět je ovlivňován nejen samotným hráčem, ale i časem.

- ⊖ změny můžou být neočekávatelné
- ⊖ časové okno (za 3 dny večer)
- ⊕ svět začíná "žít"
- ⊕ svět působí věrohodněji
- Role hráče se může zdát jako umenšená, pořád je ale klíčová

Jinak svět může působit třeba takhle: EpicNPC

# Evolving Independent World

- Vyvíjející se svět je ovlivňován nejen samotným hráčem, ale i časem.
- Ke své změně však hráče *nepotřebuje* a často si vystačí i sám.
- Svět je v omezené míře simulován.
  - Reálně
  - Statisticky
- Sandbox (Babylon[Lau16])



# Evolving Independent World

Vyvíjející se svět je ovlivňován nejen samotným hráčem, ale i časem a sebou samým.

## Změny ve světě

- pozice postav ve světě (cestující obchodníci, lodě, ...)
- dostupnost postav
- přístupnost úkolů
- dostupnost zboží, atp.
- (mikro) ekonomika
- změny ve světě nemusí být napořád, obvykle cyklické
- změny musí správně reagovat na hráče
- ...

# Evolving Independent World

To, že se věci “dějí samy” může být i pro hráče i nepříjemné - např. dlouho nevyřešený quest se “vyřeší” sám.

Komplikované je nastavení například ekonomiky, především v hrách na ni se více soustředících ( $X^2$ :Threat,  $X^3$ :Reunion, atp.).

# Evolving Independent World

Výsledky hráčovo akcí mohou být někdy zcele neočekávané a mohou pracně vytvořený systém rozbít.

Parkan I a II (vesmírná simulace):

- Lodě létají na palivo
- Palivo se těží na planetách
- Palivo se nachází v kontejnerech na lodích i planetách
- Obchodníci směňují palivo za zboží
- Zboží se vyrábí dle potřeby
- Zboží se bere ze zničených lodí pirátů
- Piráti se vytváří na základnách
- Základny je možné zničit

# Evolving Independent World

Výsledky hráčovo akcí mohou být někdy zcele neočekávané a mohou pracně vytvořený systém rozbít.

Parkan I a II (vesmírná simulace):

- Lodě létají na palivo
- Palivo se těží na planetách
- Palivo se nachází v kontejnerech na lodích i planetách
- Obchodníci směňují palivo za zboží
- Čím je zboží méně, tím je dražší
- Čím je zboží více, tím je levnější
- Když hráč prodá obří množství vybavení, jeho cena prudce klesne
- Obchodník chce prodat, nemá dost velký výdělek aby odletěl, zastaví se a nemůže odletět

# (Open) World

Při vytváření (otevřeného) světa jsou v zásadě dva možné přístupy:

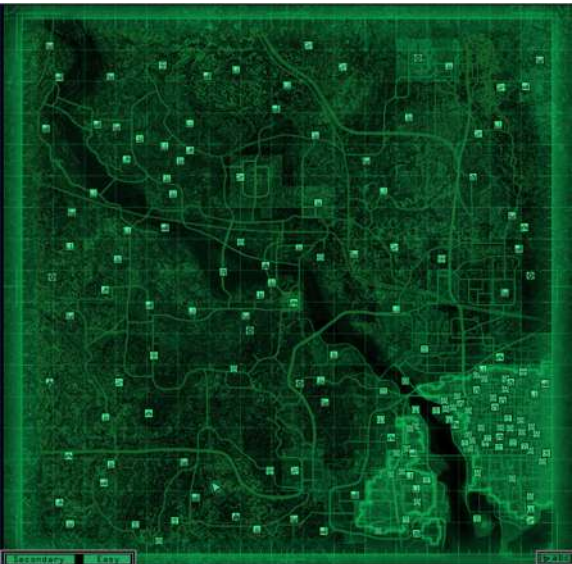
- 1 Otevřený, propojený svět v kterém jsou rozesety města, jeskyně, střetnutí a další zajímavé lokace (ala Elder scrolls)
- 2 Shluk několika oblastí (modulů, scén), navzájem značně odlišných vizuálně i příběhem (ala Baldurs Gate)

# Open world

## Propojený svět

- Umožňuje přirozenější umístění objektů ve světě
- Více immersivní, realistický
- Umožňuje lehčí vytvoření “velkého příběhu”
- Mohou (musí) vzniknout rozsáhlá “nárazníková” pásma (Fallout 3)
- Lehčí po stránce vývoje
- Struktura tvorby je **svět + střetnutí**

# Open world



# Open world

## Shluk modulů (patchwork)

- Umožňuje vytvoření zvláštních narativů pro každou oblast
- Lokálnější, ale komplexnější příběhy
- Mezi oblastmi je typicky netriviální přechod, existují hranice
- Triviální přechod do drasticky jiných terénů
- Náročnější po stránce vývoje
- Struktura tvorby dána strukturou a podobou jednotlivých **modulů**



# Konec

Dotazy?



Hello worley.  
online, 2011.



Joris Dormans and Sander Bakkes.

Generating missions and spaces for adaptable play experiences.  
*IEEE Transactions on Computational Intelligence and AI in Games*, 3(3):216–228, 2011.



Andrew Doull.

The death of the level designer.  
*ASCII Dreams: A roguelike developer's diary*, 2008.



Why is this procedural content generation while that other isn't?

In *Game Development Stack Exchange*.



Mark Hendrikx, Sebastiaan Meijer, Joeri Van Der Velden, and Alexandru Iosup.

Procedural content generation for games: A survey.  
*ACM Trans. Multimedia Comput. Commun. Appl.*, 9(1):1:1–1:22, February 2013.



Georgios N. Yannakakis a Julian Togelius Johnson, Lawrence.  
Cellular automata for real-time generation of infinite cave levels.

*In Proceedings of the 2010 Workshop on Procedural Content Generation in Games, PCGames '10*, pages 10:1–10:4, New York, 2010. ACM.



Daniel Laube.

*Babylon: Samoorganizující se parametrický model starověkého města.*

Bakalářská práce, České vysoké učení technické, Fakulta informačních technologií, Praha, 2016.



Aristid Lindenmayer.

Mathematical models for cellular interactions in development ii. simple and branching filaments with two-sided inputs.  
*Journal of theoretical biology*, 18(3):300–315, 1968.



Paul Martz.

Generating random fractal terrain [online].

[cit. 2017-12-16].



Maxim Mozgovoy.

*Algorithms, Languages, Automata, and Compilers: A Practical Approach: A Practical Approach.*

Jones & Bartlett Learning, 2009.



Procedural content generation wiki.

2019.



Yoav IH Parish and Pascal Müller.

Procedural modeling of cities.

In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 301–308, 2001.



Jan Rekers and A Schurr.

A graph grammar approach to graphical parsing.

In *Proceedings of Symposium on Visual Languages*, pages 195–202. IEEE, 1995.



Noor Shaker, Julian Togelius, and Mark J. Nelson.

*Procedural Content Generation in Games: A Textbook and an Overview of Current Research.*

Springer, 2016.



Petra Svickova.

*Parametricky vygenerovaný 3D model hradu.*

Bakalářská práce, České vysoké učení technické, Fakulta informačních technologií, Praha, 2019.

# Bibliography