

# Animace CG charakterů

---

ŠTĚPÁN KMENT



## PRÁVNÍ ZÁLEŽITOSTI

Zdroj videí – Youtube, použito v souladu s Direktivou Evropského parlamentu 2001/29/EC čl. 5(3) – fair use, illustration for teaching.

Copyright náleží majitelům copyrightu videí, kteří jsou tímto s díky uznáni.

## ANIMACE POSTAV – JAK TO BYLO BEZ KŘEMÍKU...

Postupů je celá řada a často jsou i dnes vizuálně nepřekonatelné:

- Stop-motion animace – fyzické loutky nebo modelína, krok po kroku se pohybují ručně a film vzniká postupným focením.
  - Dnes zejm. Aardman: geniální Ovečka Shaun, fotí na DSLR; pusa je element z 3D tiskárny.
- Kreslená animace – řada průhledných blan nebo průklepových papírů, každá obsahuje vrstvu scény, jdou zaměňovat. Animace se vytváří pomocí střídání blan / listů.
- Poprvé princip klíčového snímku, keyframe – hlavní výtvarník kreslí jen každý n-tý podstatný snímek, tým kolegů pak dokresluje ty ostatní, které interpolují pohyb.
  - Rumcajs, Krtek, klasické Disneyovky...

## ANIMACE POSTAV – JAK TO BYLO BEZ KŘEMÍKU...

- Rotoscoping – nafilmovali se živí herci a následně se použijí v podkladu pro kresbu.
  - např. film Alois Nebel a Pán prstenů Raplha Bakshiho (1978)
- Plošková animace – řada objektů, ze kterých je postava poskládána – hlava, trup, končetiny. Animace vzniká pohybováním objektů ve scéně.
  - např. Rákosníček
- Sestavování stavebních prvků – filmečky ve scénách z Lega

**Základní problém – problematické úpravy, pro loutkovou animaci nezbyvá než animovat od startu do konce, není možné “undo.”**

## DOBA KŘEMÍKOVÁ

- V zásadě umožňuje řadu klasických animačních technik dělat snáze s možností dodatečných oprav a úprav
- Pokrok nastal s růstem výkonu CPU / GPU a rozvojem technik pro modelování, simulace a rendering. Nejde jen o hrubý výkon procesorů, ale hlavně o objevy v oblasti algoritmů a souvisejících fyzikálních a matematických problémů (např. tak základní věc jako Bézierovy křivky!)
- Počítačové hry a interaktivita – úplně nové médium dříve v podstatě nemožné (výjimka – Kinoautomat...), “on demand” přístup, prezentace ovládaná v reálném čase.

Dále se budeme věnovat jen počítačem podporované animaci, zejména pro charaktery. Nebudeme rozebírat algoritmy, spíše praktická úskalí a problémy implementací, na které narazíte – snažte se přece dělat počítačové hry 😊

Omluvte anglické termíny v českém textu. Všichni je používáme  
a snad bude “face” přijatelnější než “ploška.”

## OBSAH PŘEDNÁŠKY

### A. Reprezentace modelů, set-up

- poloha objektu v prostoru, interpolace
- hierarchie, kostry, skinování
- jointy, blend shapes, point caches
- FK vs. IK
- rigování

### B. Zdroje animačních dat

- motion capturing
- simulace

## A. Rezentace modelů, set-up

---



## POLOHA OBJEKTU V PROSTORU

- TRS – translace, rotace, scaling: S jen výjimečně.
- T – 3D vektor – triviální
- S – 3D vektor + evnt. offset pivotu: problém s děděním scalingu v hierarchii
  - Scaling ovlivní i všechny childy!
  - Proto evnt. na childy aplikovat inverzní scaling, kompenzace.
- R – 3D vektor + evnt. offset pivotu: problém s reprezentací cpu / člověk.
  - Transformační matice – hlavou neuchopitelné, mizerně interpolují
  - Quaterniony – hlavou neuchopitelné, interpolují skvěle, výpočetně drahé
- Eulerovy úhly – yaw, pitch, roll – ok pro člověka, ale
  - Selhávají na pólech,
  - Interpolace problematická (protože po složkách),

## POLOHA OBJEKTU V PROSTORU

- Eulerovy úhly (pokr.)
  - Chaos v pořadí (XYZ, ZYX, XZX...),  
in/extrinsic ref. frame – nejednoznačné
  - Eliminace např. roll rotace – jen pokud je poslední v pořadí!
- Realistické řešení: v GUI eulerovy úhly, interně quaterniony, při renderingu převod na trans. matice.  
Data reprezentovat jako TRquat nebo TRquatS
- Pivot: musí být korektně nastaven, jinak selže interpolace mezi keyframes. Prakticky např. řešit vložení jointu do hierarchie na místo pivotu jako parentu daného objektu.

## INTERPOLACE

- Obvykle má každý parametr buď konstantní hodnotu, nebo kontejner s klíči, které definují v čase křivku. Klíče lze interpolovat různými způsoby, u rotace se navíc musí interpolovat všechny složky naráz v jednom kroku.
- Typicky bézierovy křivky nebo TCB – hladká křivka: handles / TCB parametry řídí průběh interpolace, např. zrychlení. Za normálních okolností je v animaci vždy nutno dodržet nástup akcelerace a decelerace, jinak animace začne vypadat “upatlaně.”
- Lineární interpolace – atypická, ignoruje akceleraci, objekt se skokem dostane na konstantní rychlost. Ale výpočetně velice levná, např. interpolace barev?
- Step, konstantní – drží hodnotu do dalšího klíče: např zap/vyp viditelnost aj.

## INTERPOLACE

- Pro simulaci reality musí být spojitá jak křivka polohy, tak její 1. derivace – rychlost, ideálně i 2. derivace – zrychlení; jinak vede k vizuálním defektům! Platí i pro rotaci vyvolanou např. svařem.
  - Výjimka – náraz: do zdi, pata do podlahy. Jinde začne sval nebo motor vyvíjet sílu, ta postupně podle hmoty rozpohybuje těleso.
  - Problém stříhu kamery ve filmu: ve stříhu se scéna skokem může změnit, jak zabránit snadno interpolaci, pokud exportér nepodporuje značku pro stříh?

## HIERARCHIE, KOSTRY

- Máloco je opravdu volný objekt, většinou hierarchie: auto > karosérie > dveře > okno. Hierarchie parent > child, nejvyšší parent hierarchie je root. Objekt kostry, kloub, se označuje jako joint.
- Model člověka: root bývá pánev. I ve skutečnosti je těžiště člověka cca. v okolí pupku. Při úderu do břicha letí celé tělo pryč, úder do dlaně ovlivní jen ruku.
- Pro zjednodušení lze říci, že kostra ovlivňuje celkový vzhled těla, pro grafiku ve hrách to může stačit. Jenže jde o velice hrubou aproximaci!
  - Ale: ne všude jsou kosti jako třeba pánev – stehno. Lopatka / klíční kost není kloub v jamce, podobně předloktí není jediná kost: nutno aproximovat.

## HIERARCHIE, KOSTRY

- Páteř se v tlaku podstatně komprimuje! Markery mocapu kompresi uloží, ale kostra se statickými délkami kostí řízená markery se může nevhodně kroutit.
- Problém stehna – kost má pivot otáčení mimo osu. Pak segment CG lýtka nemá být rovnoběžný se segmentem stehna pro vhodnou aproximaci, to ale rigy nepodporují...
- Kůže je v realitě navázána na kosti, ale nepřímo kvůli vrstvám svalů: to je problém v ultra-realistické grafice: např. na stehnu koně pohyb kůže s kostí souvisí dosti vzdáleně, podobně obočí člověka klouže po lebce. Řešení: virtuální kosti ovládající dané kousky kůže, nebo úplně jiný přístup (viz dále).
- Typicky root má animované TR, childy mají statický lokální T offset a mění se jen R.

## HIERARCHIE, KOSTRY

- Aby R byly rozumně popsány a daly se komprimovat, kostra se tzv. resetuje: dodatečným vektorem pre-rotace se joint otočí, aby jedna z os (+X) mířila k dominantnímu childu, druhá reprezentovala hlavní osu otáčení (loket...). Prerotace je statický R vektor.
  - Stejně nefunguje vždy, viz roll lokte při úplném propnutí – nečekaná další rotace... (také lepíte kostru na markery v T póze?)
  - Tj. joint se vůči parentu posune (lokální statická T – bez podpory komprese páteře...), prerotuje (statická R), rotuje (animovaná R).
  - Resetování vhodně provedené umožňuje i zrcadlení póz mezi levou a pravou částí těla, tj. rozmyslet, jak + rotace vpravo má vypadat vlevo (např. pro ramena z rozpažení do předpažení vpravo má totéž udělat vlevo, skrčení prstu vpravo totéž vlevo); vhodně řešená prerotace to umožní.

## HIERARCHIE, KOSTRY

- Dominantní ohyb je jen na jedné / dvou křivkách R (záleží na stupních volnosti, DOF, kloubu). Pokud se lze vyvarovat pólů v eulerových úhlech, je to nejlevnější řešení, evnt. kompletně některé křivky ignorovat nebo podsamplovat. Pro rameno a kyčel není možné, raději quaterniony.



## SKINOVÁNÍ NA KOSTRU

- “Nalepení” meshe na související kosterní strukturu.
- Velká alchymie, začátečník dosáhne maximálně vzhledu gumové hadice.
- Nejhorší je asi kyčel, rameno. Obtížné jsou i kývač hlavy, koleno, loket, kotník... čím dramatičtější ohyb a volnost kloubu, tím více problémů.
- Skinuje se často v A póze, aby mesh v nativním stavu bez deformací byl v průměru typického vzhledu (postavy chodí, běhají, evnt. mírně zvedají ruce).
- Protože od A ruka zvednutá nad hlavu v rameni udělá i  $150^\circ$  odchylku(!), používají se pomocné jointy řízené pomocí matematického vzorce (expression), které sdílejí rotaci napůl. Mesh se pak snáze naváží na více jointů a zabrání se tak masivním deformacím např. podpáždí.

## SKINOVÁNÍ NA KOSTRU

- Podobně problém roll, podélné rotace na předloktí, více jointů rozkládá rotaci a brání zaškrcení meshe.
- Břicho: ve skutečném těle páteř je na zádech, při předklonu se hmota v břiše stlačuje. V CG se kostra dává spíše do středu trupu, aby ohnutí zad nezpůsobilo masivní deformace.
- Není jediný skinovací algoritmus! Navíc real time engine může omezit počet kostí, které vertex ovlivňují, jen na tři, čtyři dominantní. Pak v 3D softwaru můžete vidět jiný výsledek než ve hře.
- Skinovací algoritmy: populární
  - Blendování transf. matic – nejsnazší, s defekty
  - Dual quaternion – výpočetně drahé, ale omezuje ztráty objemu meshe při ohybech (mimořádně, český vynález Ladislava Kavana, FEL ČVUT)

## SKINOVÁNÍ NA KOSTRU

- Poloha CG jointů v meshi JE podstatná, špatně pozicovaný joint deformuje mesh při větším ohybu tak, že není možné problém nijak kompenzovat na úrovni animačních dat! Např. zápěstí o pár cm blíže k lokti = katastrofa.

## JOINTY, BLEND SHAPES, POINT CACHES

### Jointy

- Pomocí jointů a kostry se dobře animuje hlavně trup těla do celkové globální pózy. Na nuance se nehodí.
- Nejsou vhodné pro kůži, která klouže na kostech, např. mimika tváře.
- Pletence svalů lze aproximovat mnoha jointy a jejich scalingem, pokud to engine ale podporuje.
- A vůbec nejdou použít na dynamicky se morfující objekty bez pevného tvaru.

## JOINTY, BLEND SHAPES, POINT CACHES

### Blend shapes (morph targets)

- Řada meshů se shodnou topologií a odlišným vzhledem. Tj. v meshi naposunujeme vertexy, ale neměníme jejich počet. Obvykle se pak i prolínají textury či wrinkle mapy.
- Meshe lze mezi sebou plynule prolínat a následně ohnout jointy.
- Vhodné pro pletence svalů a hlavně mimiku, na druhou stranu neumožňuje tento přístup lokální kontrolu nad meshem, proto špičkové modely kombinují oba přístupy.
- Jednotlivé targety obličeje často vychází z FACS Paula Ekmana, v principu jak svaly na hlavě dovedou kůži deformovat – mrknutí, otevřená ústa, povytažené obočí aj.
- Získat targety lze fotogrametrií (Agisoft Photoscan) a potom projekcí jednoho meshe s vhodnou topologií na jednotlivé scany (R3DS Wrap)

## JOINTY, BLEND SHAPES, POINT CACHES

### Point caches

- Volně v čase animované meshe. Velmi mnoho dat, která se dají mírně komprimovat, mohou reprezentovat jakékoli změny tvaru. Ve hrách např. CryEngine – Ryse: trhající se plachty lodě.
- Typicky spíše výsledek nějaké fyzikální simulace, k dosažení je třeba velmi mnoho výpočtů; ve hrách asi poněkud “high end.”
- Změny topologie, polohy vertexů, evnt. UV texturových koordinátů...
- Formát dat Alembic – používá se pro přenos scény do rendereru aj., vhodný i pro tento účel.

## FK VS IK

- Forward, inverse kinematics, všichni znáte princip: ohýbám postupně od rootu kosti vs. chytanu koncový joint a hierarchie se naopak přizpůsobí směrem k parentu
- V realitě to je podobné: volná končetina se pohybuje přes FK, obecně po obloucích (ruce v chůzi aj.), ale pokud je noha v kontaktu se zemí, působí zpětně na hierarchii směrem k rootu; podobně opření s propnutýma rukama o stůl.
- Odlišná interpolace: FK je interpolace rotací, vede k přirozenému pohybu; interpolovaný IK efektor se pohybuje po úsečce, vede k neobvyklému pohybu (mytí okna houbou).
- Proto se vhodně mezi FK a IK přepíná, v Autodesk MotionBuilderu jsou dva setupy a mezi nimi lze blendovat dle potřeby.

## RIGGING

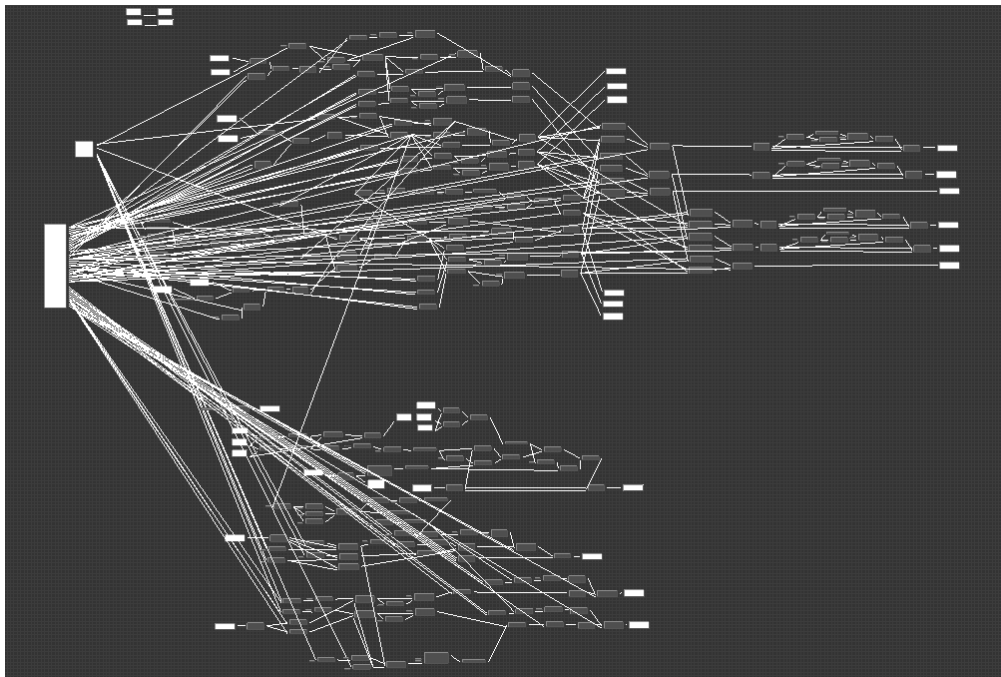
- Pokud chcete jako technici dělat ve filmu, toto by vás mohlo velice zajímat! Pro lidi s vizuálním citem, hračky a trochu programátory.
- Ovládací setup, kterým animátor postavu rozumně může ovládat. Bez něj by musel kostru ohýbat FK postupem kost po kosti.
- IK setup pro ruce, nohy, evnt. celé tělo (tažení dlaně napne ruku a pak dále začne ohýbat i trup), setup pro přenos motion capture dat, vstup křivek fonémové analýzy aj.
- Setup “kontrolék,” sestavy objektů, kterými se rychle dá postava snadno a s predikovatelným výsledkem ovládat.
- Často se pomocí expressions, vzorců, řídí pomocné struktury - např. podle dráhy kola se odvodí jeho rotace, podle úhlových akcelerací kostí se zvýrazní svalovina atp.
- Setup pro chlupy, vrásky.



## RIGGING

- Problém: “knee /elbow snapping”: nepatrný nárůst (3 mm...) vzdálenosti kotníku a kyčle při téměř propnuté noze způsobí drastickou změnu polohy kolene až do saturace a prudkého propnutí. Protože noha musí logicky zůstat na zemi, je nutno místo toho posunout vhodně kyčel, zarotovat pánví nebo i posunout celou páteř.
- Problém – kontrolky nad morph targety – pokud mají sledovat mesh a dodatečně jej měnit, musí umět sledovat jednotlivé vertexy v meshi, MotionBuilder to neumí.
- Sofistikované rigy zejm. v Autodesk Maya, ale lze vybudovat téměř v jakémkoli 3D SW.

## RELATIONS – PRO PŘEDSTAVU...



## B. Zdroje animačních dat

---

## RUČNÍ ANIMACE

- Podobně jako klasická animace je jen pro mimořádně talentované **umělce**, nikoli techniky. Každý jsme “fotograf,” “sazeč” (nejlépe ve Wordu), “animátor,” ale k dokonalosti je dlouhá cesta.
- Nutný cit pro dynamiku pohybu, cykly chůze...
- V ČR mnoho elitních animátorů nenajdete, jsou dávno za kopečky.
- Vidět lidi od Disneyho je doopravdy zážitek.  
Podívejte se také na making of Ovečky Shaun...
- Je to jako malování obrazu, my uděláme jen patlanici, umělec udělá obraz, ohne realitu podle potřeby do zajímavého výsledku.

## MOTION CAPTURING

- Jako fotoaparát, teoreticky rychlejší a levnější než ruční animace. Pro zachycení reality není třeba dnes krajinu pracně kreslit, snazší je ji vyfotit. Filmy Pixaru tím neuděláte, ale realistický pohyb digitálního dubla ano. Proto se s klasickou animací nevyklučuje, focení také nevyhubilo malíře.
- Je třeba pokud možno nahrát vše naráz - pohyby tváře nahrané odděleně od těla nikdy nebudou dostatečně synchronní. Jsme léty trénování ve čtení jemných nuancí v pohybu a hlavně výrazů tváře, proto je velký problém je napodobit! Bohužel neexistuje žádná technologie pro všechno, proto je performance capturing tak složitý.

## MOTION CAPTURING

- Uncanny valley efekt – jak se blíží CG dokonalosti, náhle divák přestane postavu vnímat jako animovanou, ale jako nedokonale skutečnou – ta pak vypadá odpudivě, “creepy” – např. Polar Express Roberta Zemeckise.
  - Překonat tuto bariéru je velice náročné (90% zabere 90% času a zbylých 10% zabere dalších N x 90% času 😊)
- MC je kritizován za deformování pohybů. To jistě je pravda, markery nebo senzory na těle se totiž nepohybují tak, jako kosti pod nimi, navíc hlavně CG kostra neodpovídá skutečné – i když se proporcemi třeba blíží, stejně přenos z MC dat dá pohybu “hodně zabrat.”

## MOTION CAPTURING

- Část vývoje mocapu směřuje právě do solverů, které přenesou pohyb z optických markerů na kostru nebo z jedné na druhou kostru s jinými proporcemi (retargeting). Viz např. IKinema, Calcium solver (Motion analysis); naopak Actor v MotionBuilderu je spíše špatný. Ideální retargeter zachová “esenci” pohybu a přenesese ji na úplně odlišnou kostru!
- Dnes špičkové CG charaktery prakticky nepoznáte od živých: Planet of Apes od Wety...

## MOTION CAPTURING

- Nahradí herce? V dohledné době ne, protože je daleko, daleko nákladnější. Navíc se motion capturing mění na performance capturing, kde právě o kvalitní hraní jde především.

### Performance is the king!

- Jako technici si budete rádi hrát s křemíkovými hračkami, ale na place je nejdůležitější herec a jeho projev, ne vy. Smiřte se s tím.
- Navíc ve skutečnosti se data masivně dotahují ručně, nikdo nemá MC technologii, která by hero charakteru AAA filmu udělala sama: v making of už člověkoroky animátorů nevidíte.



## POHYB TĚLA – “KLASICKÉ” MOCAPY

- “Ohlodaná kost,” už se zkrátka umí. Optický MC je sice omezen velikostí placu a je celkově velmi komplikovaný, gyroskopické systémy ale trpí velkými nepřesnostmi, které se obtížně eliminují, proto optika doteď jednoznačně vede.
- Výhoda optiky je v zamknutí vůči prostoru (herci si podají ruce = CG postavy také, CG postava se drží přesně stop herce), markery lze umístit na cokoli (zvíře, pušku, míč, bič, hrnek, pero...) a jsou levné, 10 lidí není 10x cena MC zařízení.
- Optikou nahrávané prsty: netriviální: prsty mají mnoho stupňů volnosti a masivní zákryty, vyžadují vysokou přesnost nahrávání. V praxi buď pouhý spár svírající ruku nebo setupy podpořené různými heuristikami
  - (Štěpán K. - vlastní solver vyvíjený několik měsíců)

## POHYB TĚLA – “KLASICKÉ” MOCAPY

- Optikou nahrávaná mimika – extrémně malé markery: nutnost mnoha kamer s teleobjektivy, potíže s kalibrací a udržením přesnosti. Data ve skutečnosti vyžadují masivní ruční opravy.
- Vicon Cara – malý mocap na helmě jen na obličej. Pořád ale přímo nenahrává pohyb zornice oka (velký problém!) nebo vnitřek úst.

## MIMIKA TVÁŘE

- Pomocí analýzy obrazu počítačovým viděním. Head mounted camera, HMC – 1-3 kamery na helmě pevně stažené k lebce jsou namířené na obličej. Analýzou obrazu se detekují kontury rtů, očí, středy zornic a event. značky na kůži.
- Analyzovaná data se převedou obecně na pseudo-FACS křivky, např. míra otevření úst v čase, míra povytažení obočí v čase aj.
- Následně se křivky použijí k řízení rigů. Proto ideálně rig je postaven pro daný solver – např. Cubic motion + 3lateral.
- Sledování pohybu očí – eventuálně pomocí Eye trackerů
- Nebo point caches přes fotogrametrii – L. A. Noire
- Pozn. – bez skin shaderu a kvalitního nasvětlení stejně výsledek bude mizerný.

## MIMIKA TVÁŘE

- Alternativně pomocí fonémové analýzy hlasu: analyzátor vygeneruje křivky % zastoupení fonémů v čase a tak je možno přijatelně rozpohybovat ústa.
  - Nálada se dá odhadnout maximálně z obsahu dialogu samého nebo dle hlasitosti v dané chvíli, v praxi se spíše značkuje ručně.
  - Horní půlka tváře zůstává bez emocí a je těžké ji přijatelně dokončit. Metoda je vhodná pro velké objemy dialogů bez nároku na preciznost, mrkání a nálady lze rychle donahrát pomocí myši a joysticku.
  - Proč nefunguje v multiplayeru v real time? Ústa se svaly ohnou do pózy a následkem je daný zvuk. Pokud ten analyzujeme, musíme jej nahrát, analyzovat a opět napřed ústa CG modelu dostat do pózy a zvuk přehrát: asi 1/3 sec lag.

## SIMULACE

- Ani stunt double nezahraje veškeré pády – řešením je simulace pohybu kostry.
- Výhodou je, že pády nemáme “nakoukané” a tak neposoudíme snadno, nakolik animace je či není reálná: poněkud nižší nároky na realističnost.
- Nejjednodušší je rag doll solver: odpovídá snad jen zastřelené postavě, která bezvládně padne k zemi.
- Aktivní chování v pádu – behaviors. Sofistikovaný algoritmus, který emuluje svaly postavy a vyvíjí síly způsobující pohyb kostry. Postava v pádu dá ruce před sebe a snaží se dopad utlumit pokrčením kolen jako skutečný člověk. NaturalMotion Euphoria.

## SIMULACE

- Klíčová pro hry: mocap data tvoří “animační obálku,” pokud z ní postava vypadne, např. pádem přes obecnou překážku, simulace musí postavu nějak vrátit do obálky zpět – není možno nahrát pád na každém svahu a na každé překážce, vede na nekonečno pohybů!
- Hezký případ: Assassin’s Creed (UBIsoft): tisíce nahraných pohybů, dle potřeby se pohyb dotáhne IK efektořem a postava dohmátne správně na chyt ve zdi.
- Změna animace – např. meč v ruce: dodatečná zátěž změní celou dynamiku pohybu a posune těžiště, nelze beztrestně aditivně sčítat animace (snad vyjma prstů a rotace hlavy), postava pak mečem mává jako nafukovací hračkou (kdo hrál Gothic?)

## KONEC!

Dovolte mi tady skončit. Problematika je nesmírně rozsáhlá a dalo by se pokračovat další hodiny.

\_díky za pozornost! Pojd'me na videa a dotazy.