



Human-Audio Interaction Lab
hailbringer.com

INTRO

Vorstellen

Ziel des vortrages -> Vertragsvarianten

Vortrags-Zeit: 15 min

Persönlicher Hintergrund, Musikwissen

Thema: Partitur, nicht Instrument!



klassisches recording studio



großes mischpult



produzent blickt vom control room in den recording room



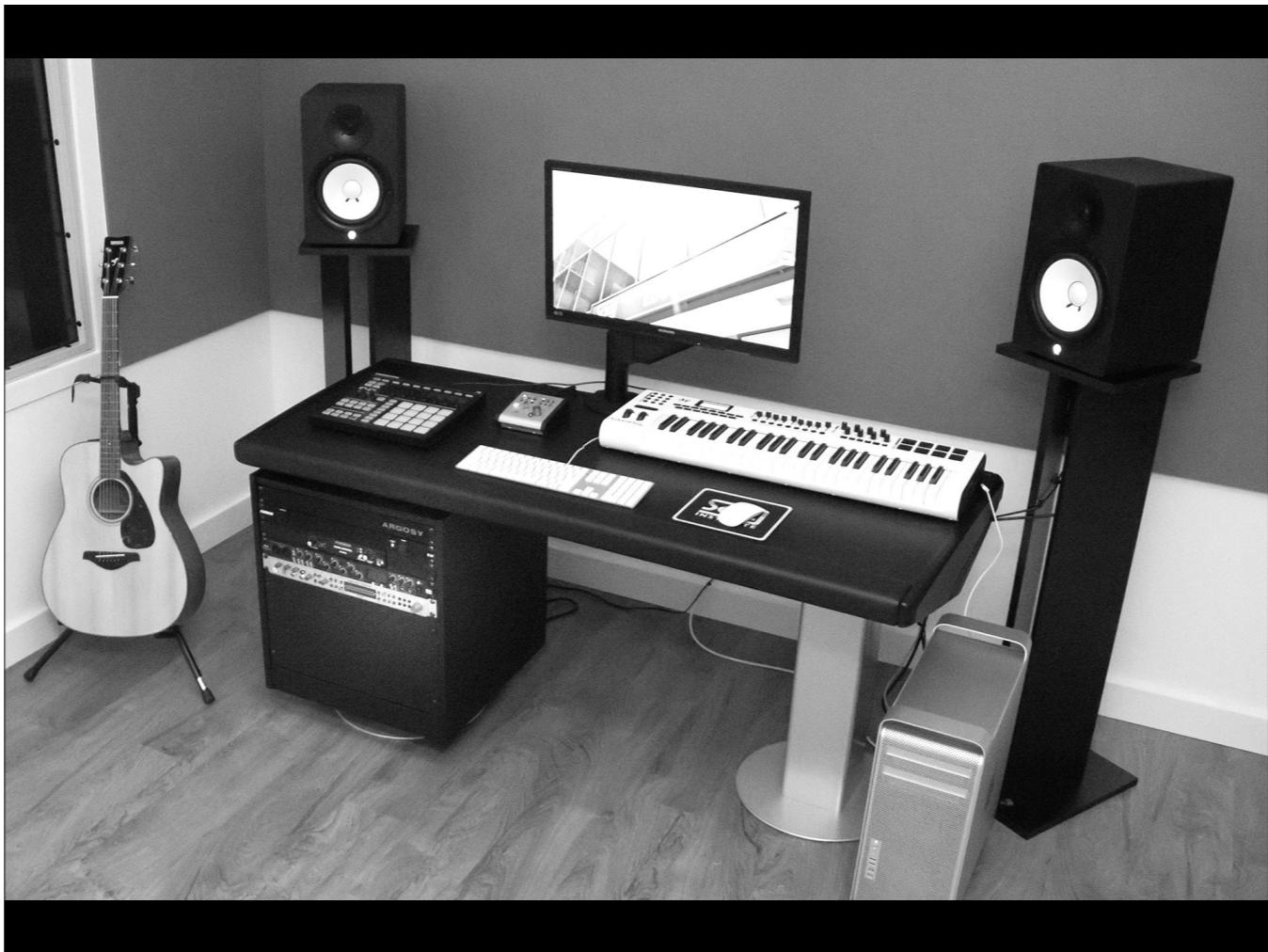
mikro-ständer, klavier



wandel: blick auf bildschirm
software im fokus



digitalisierung, verkleinerung der hardware



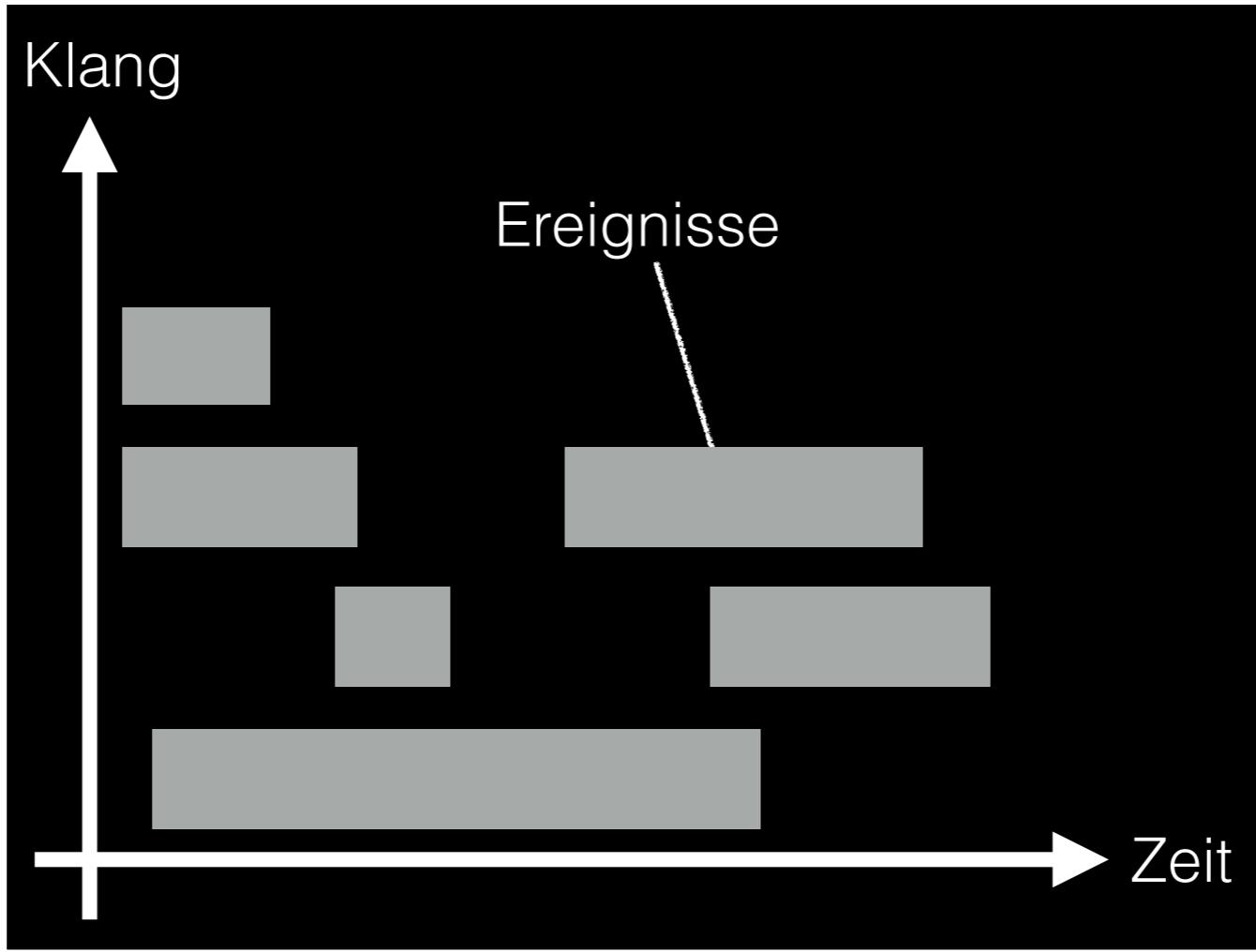
home studio



mehrere monitore!







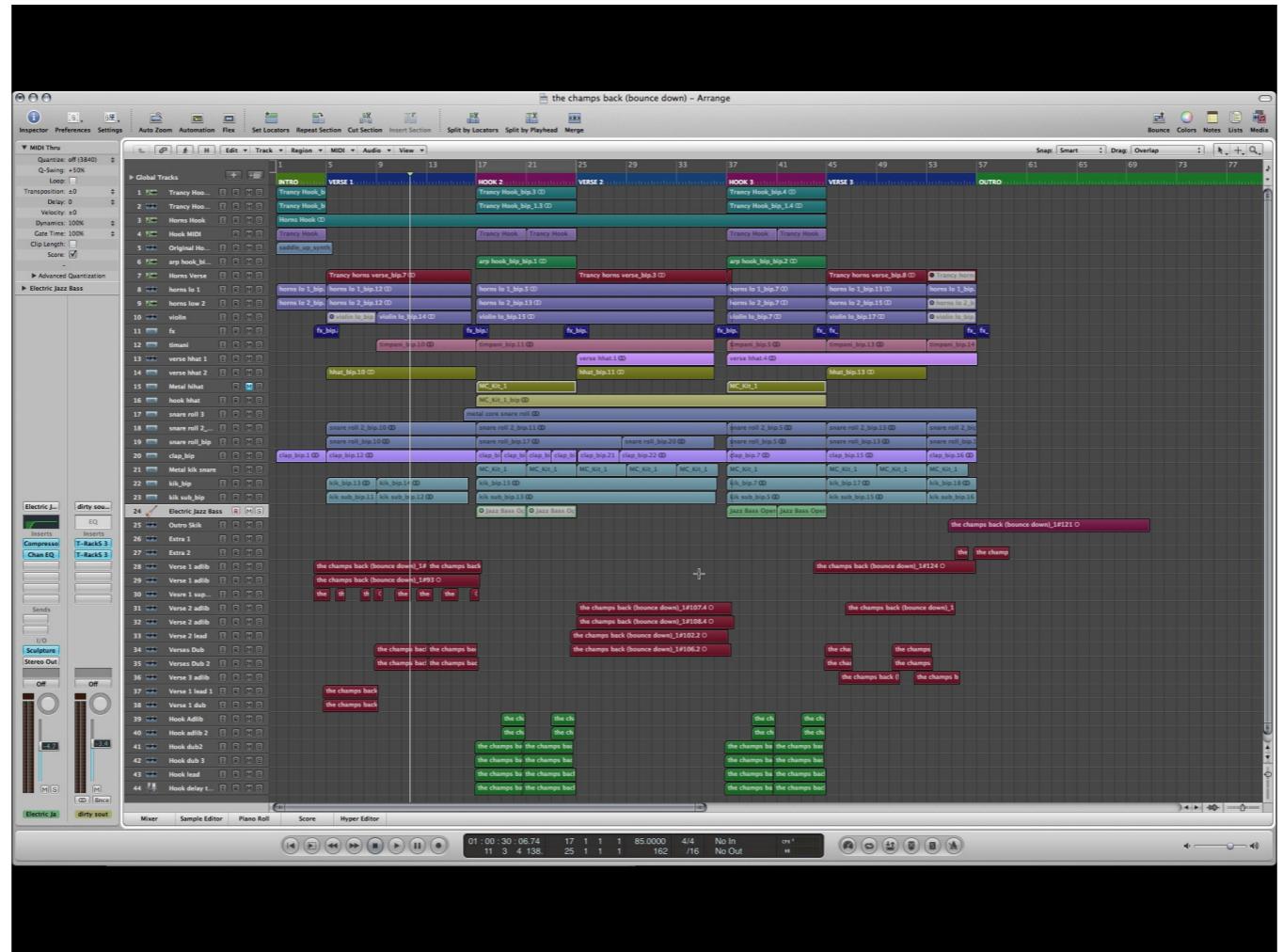
Sequenzer software und DAWs

2d darstellung

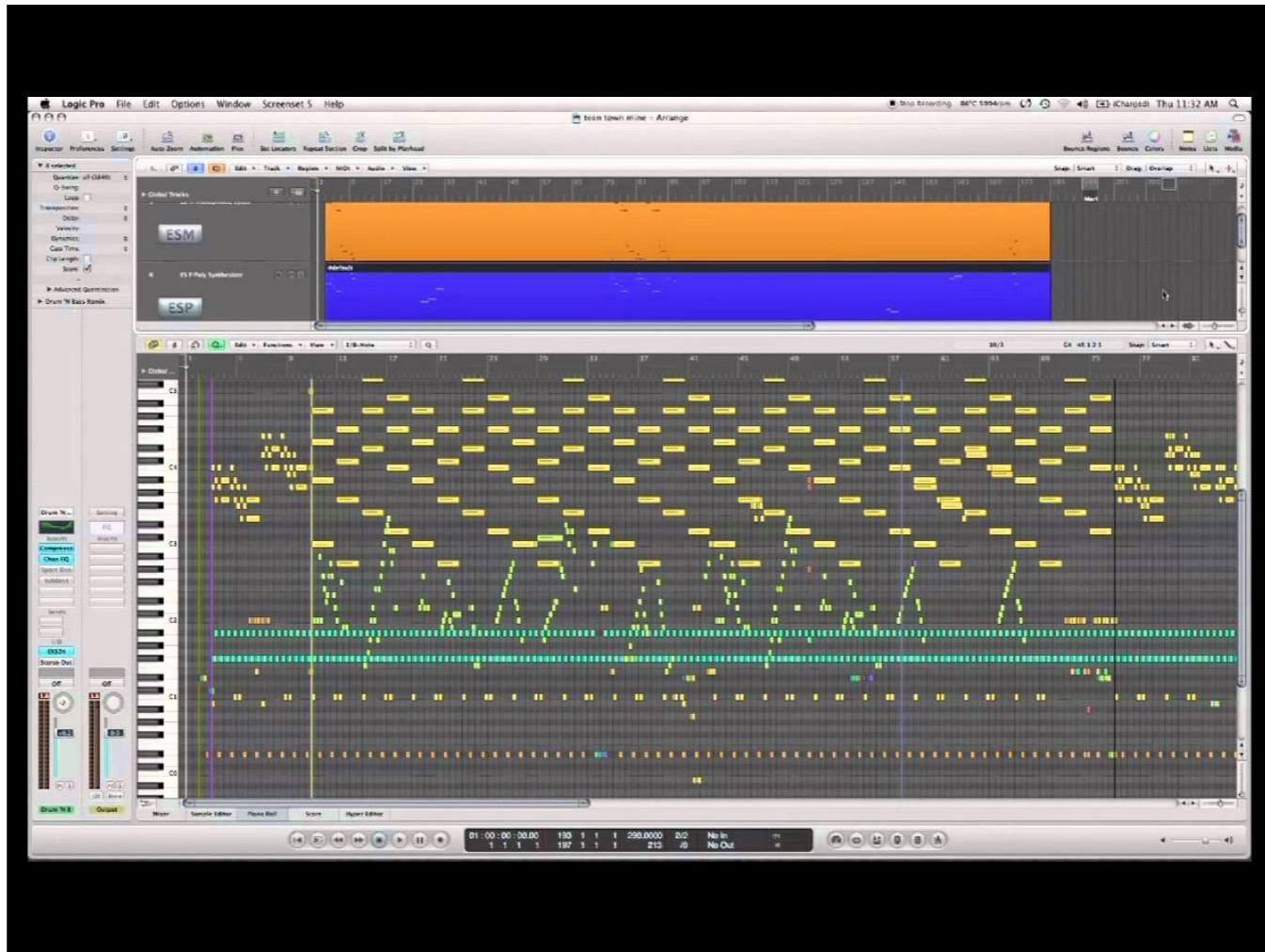
klang:

- * tonhöhe ... Töne mit parametern

- * instrument ... Instrumente mit regionen



arrange view



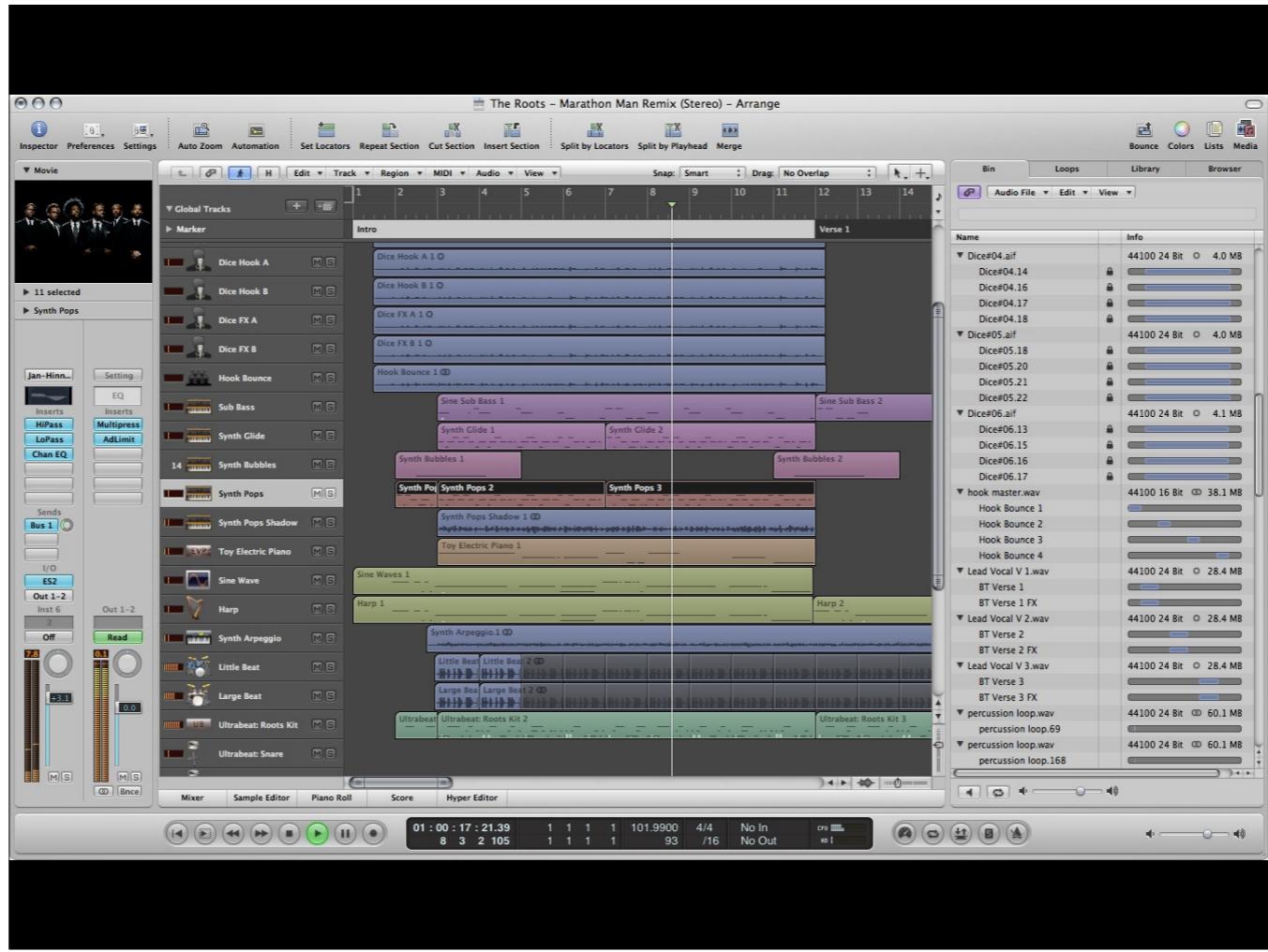
piano roll



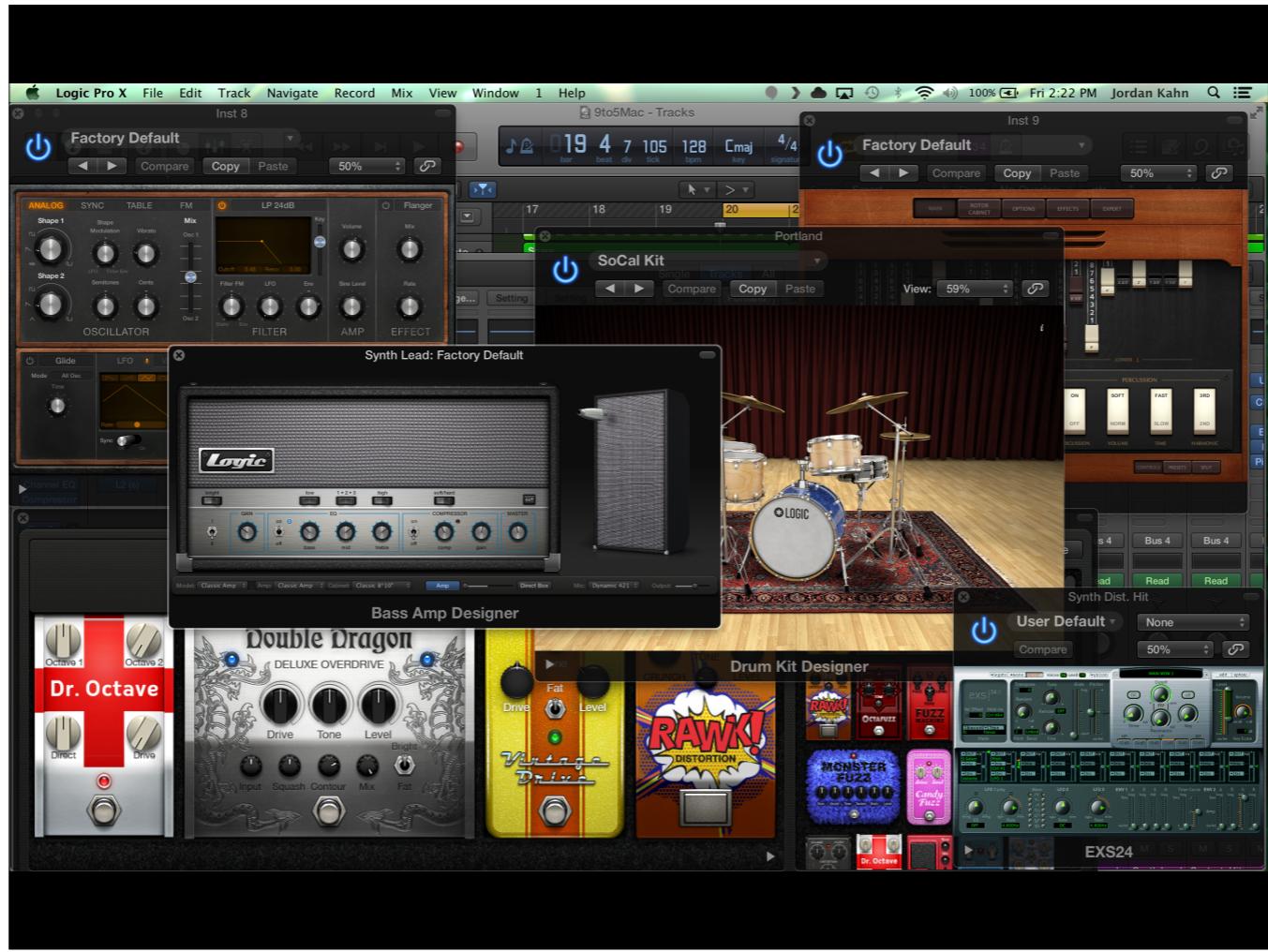
Kontexte in DAWs

was noch alles dazu gehört ...

am beispiel Apple Logic Pro ...



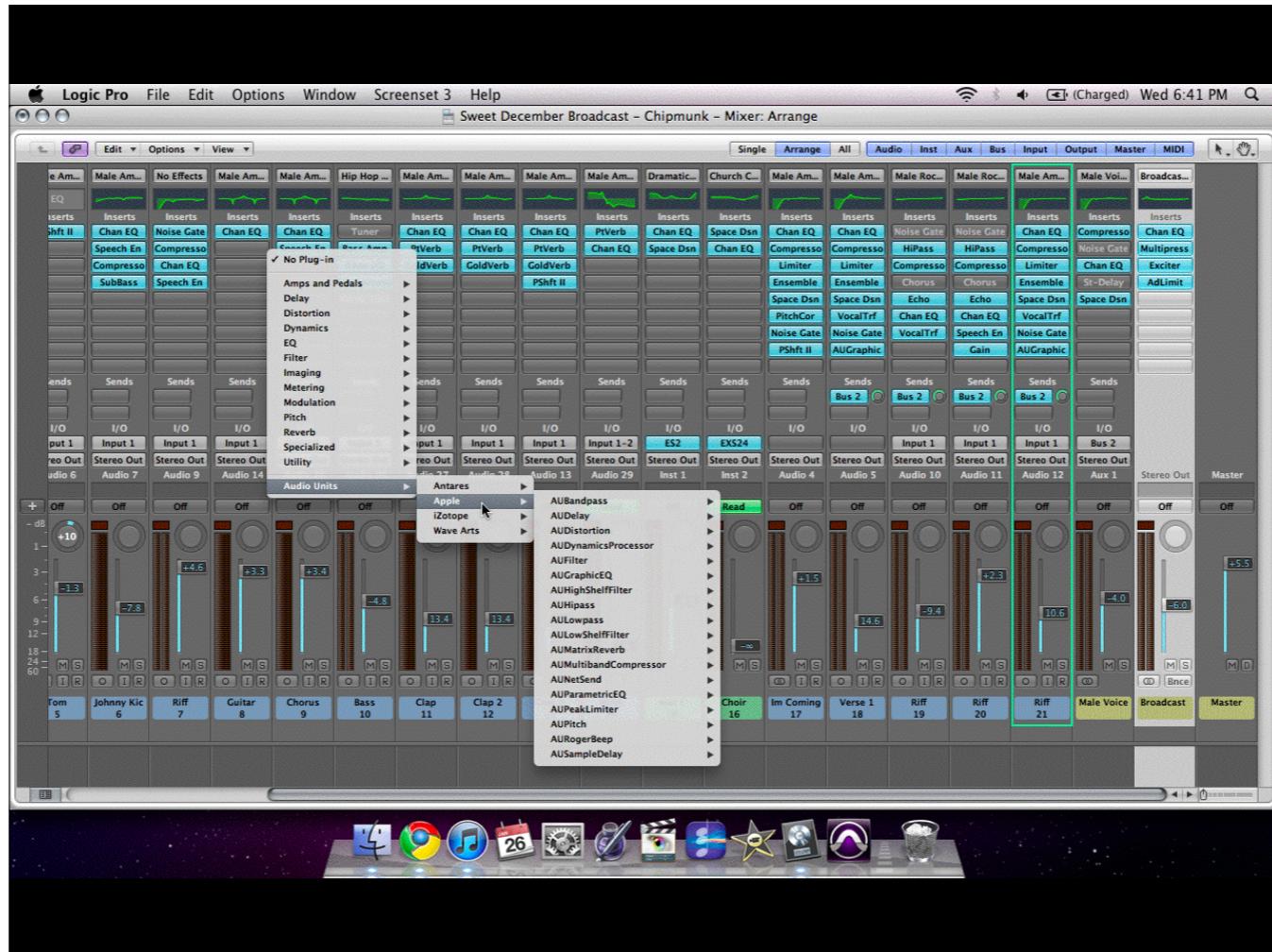
data management



instruments



effekte



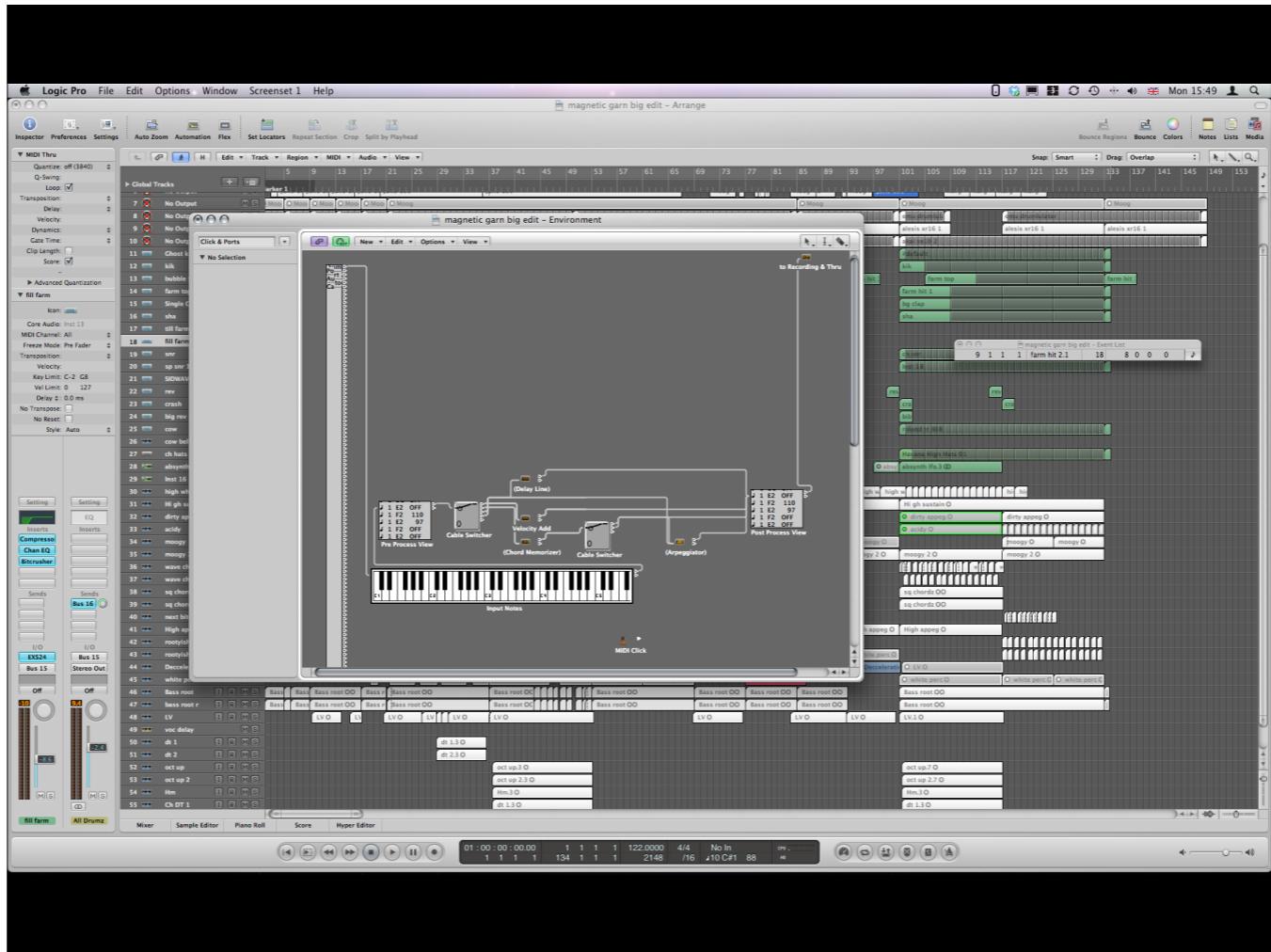
mixer



parameter



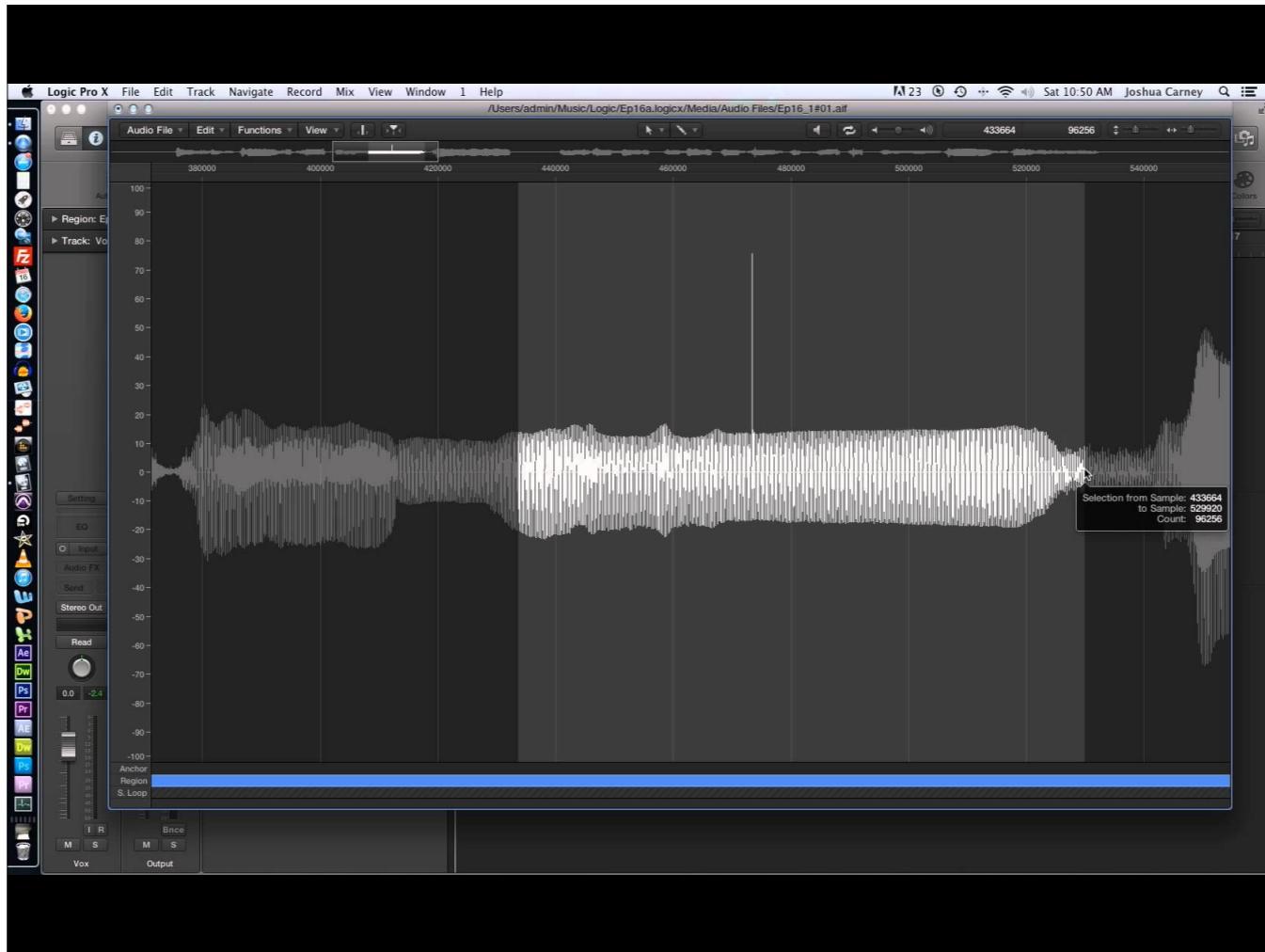
sampler



signal-wege



zusammenhang: arrange view, midi, instrument, kanal, mixer, effekt

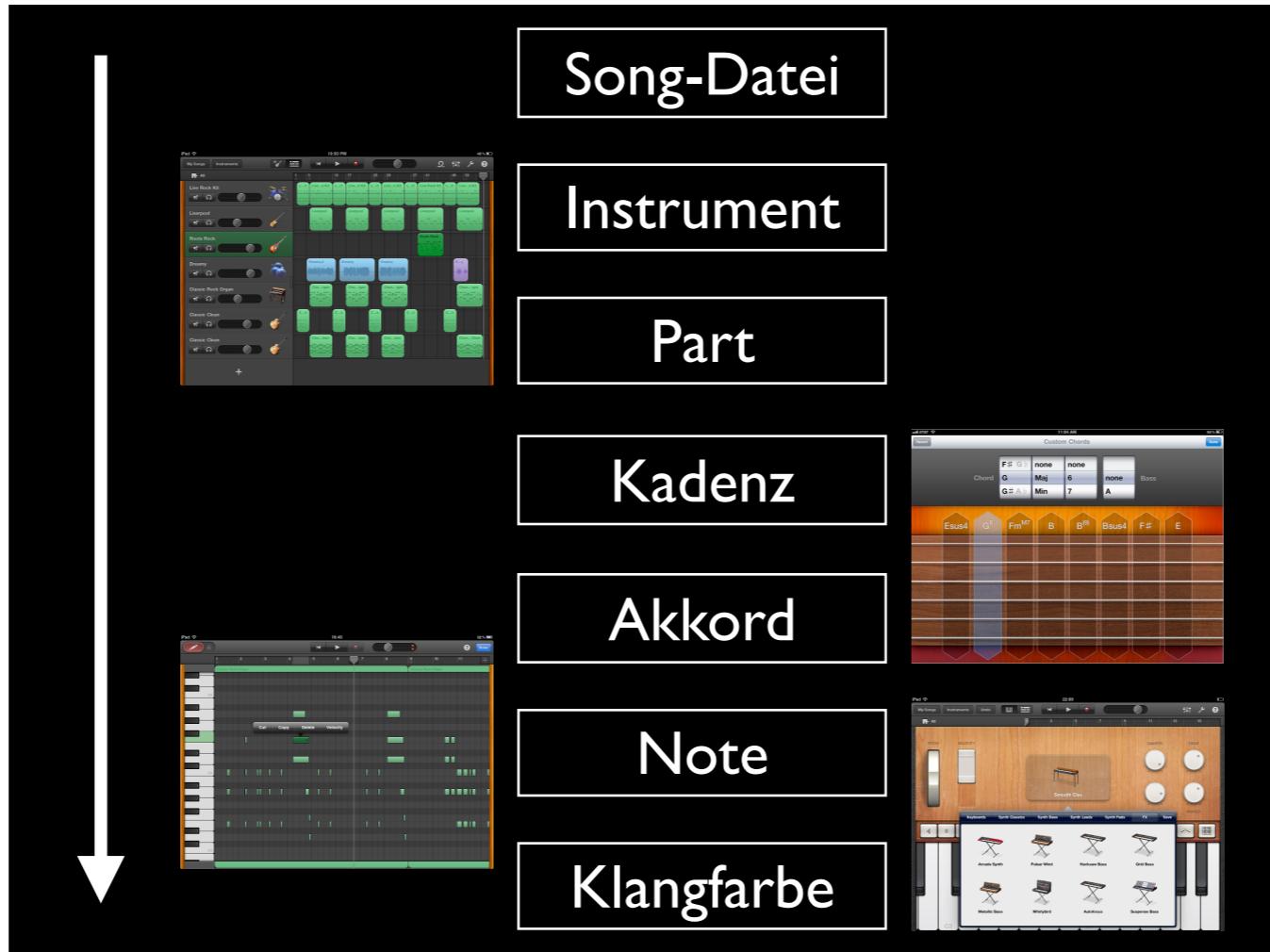


audio-bearbeitung



und tablets?

imagine logic pro instruments as ipad apps!



- different formalizations of the higher level structure of music
- but they all have one fact in common: Music is hierarchically structured
- the example is arbitrary and simplistic. there are endless ways of defining a hierarchy

List of Requirements

1	Cognitive Styles	31
2	Cognitive Dimensions	34
3	Constraints	35
4	Concentration	36
5	Accessibility	36
6	Mono-tasking	37
7	Single Interaction Context	37
8	Single Physical Mode	38
9	Scalable Graphics	38
10	Fluid Panning and Zooming	39
11	Relative Mixing	39
12	Complementarity	40
13	Interchange	40
14	Self-sufficiency	41
15	Generality	41
16	Emergent Exploration	42
17	Interface Exploration	42
18	Direct Exploration	43
19	GCDCG-Sequencer	45
20	Transient Voices	46
21	Voice Groups	46
22	Emergent Mix Groups	46
23	Juxtaposability	47
24	Groove	48
25	Temporal Abstraction	48
26	Liveness	49
27	Direct Audio Processing	50
28	Scalable Audio Processing	50
29	Rich Audio Processing	50
30	Visual Audio Processing	51
31	Nondestructive Audio Processing	51
32	Preparation	52

POTENTIALE

=====

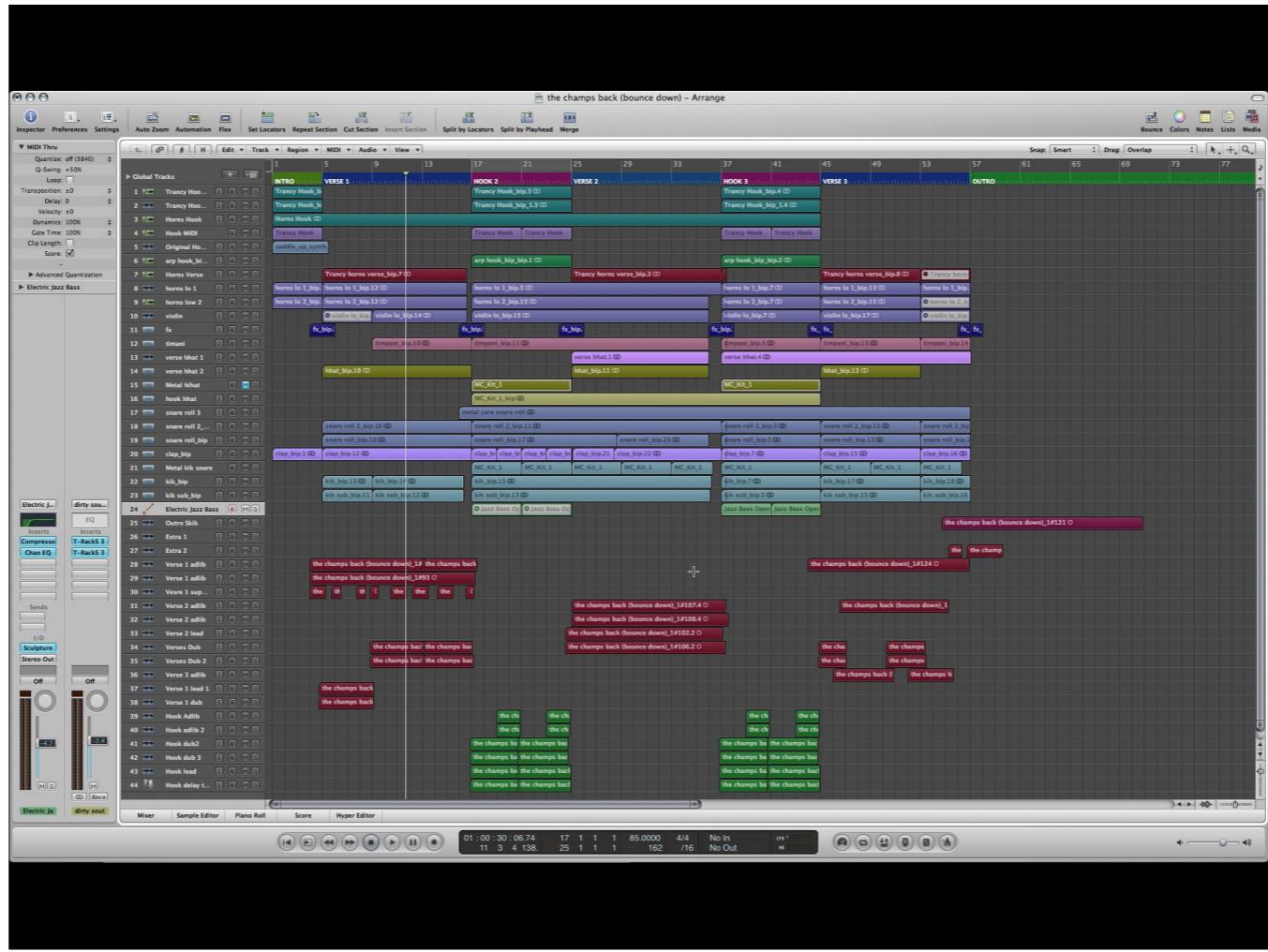
Probleme identifiziert

36 anforderungen ...

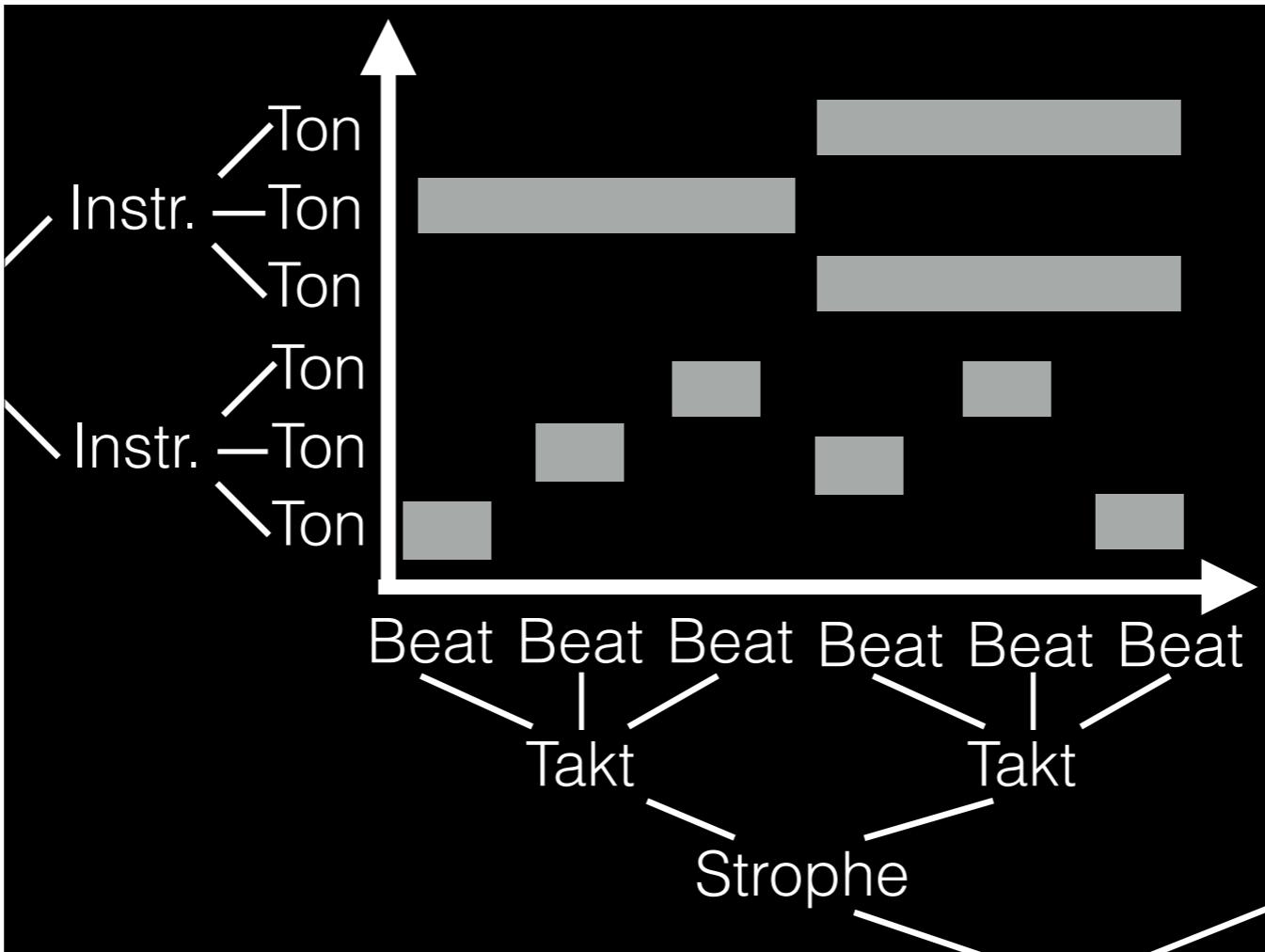
33	Global Reuse	53
34	Versioning	53
35	Referencing	54
36	Variation	54

in 4 bereichen:

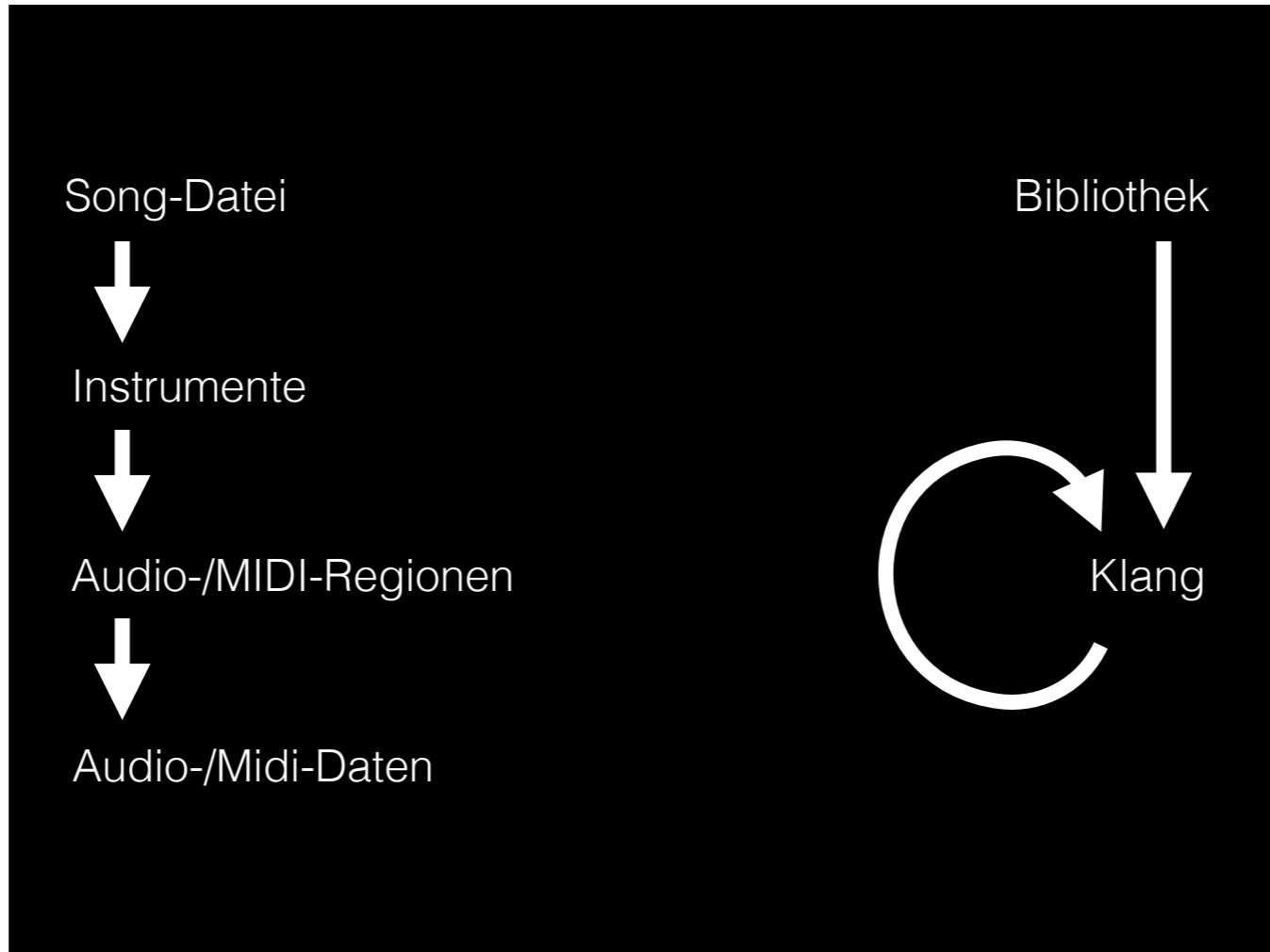
- * simplicity
- * freedom
- * exploration
- * abstraction



- * Zooming! -> Tablet
- * Scalable graphics -> Tablet

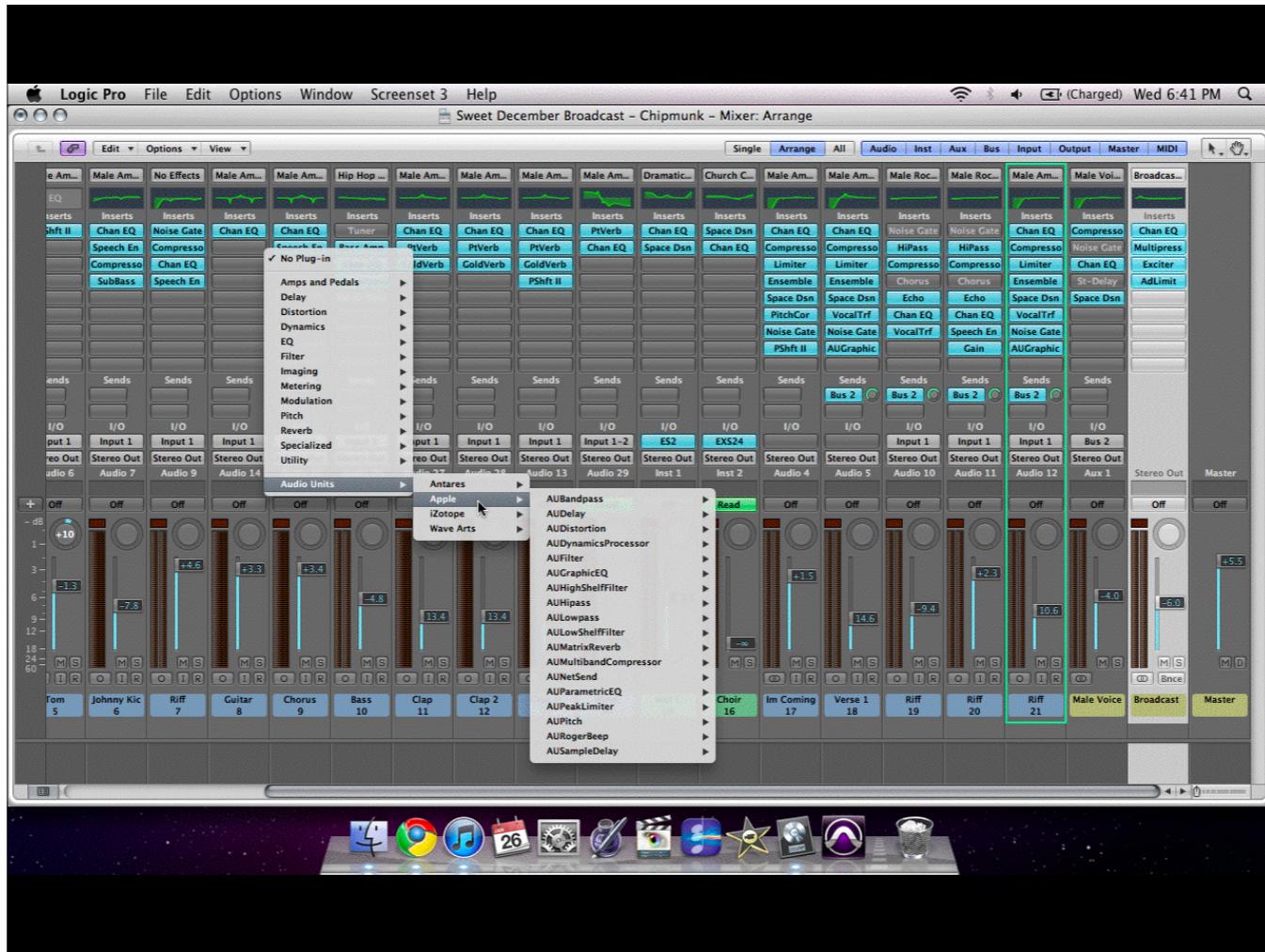


- * Gcdcg : klassische grafische kompositions-software + abstraktion + improvisation
- * abstraktion: benutzerdefinierte konzepte! beliebige verschachtelungstiefe
- * semantisches zoomen der zeit-achse

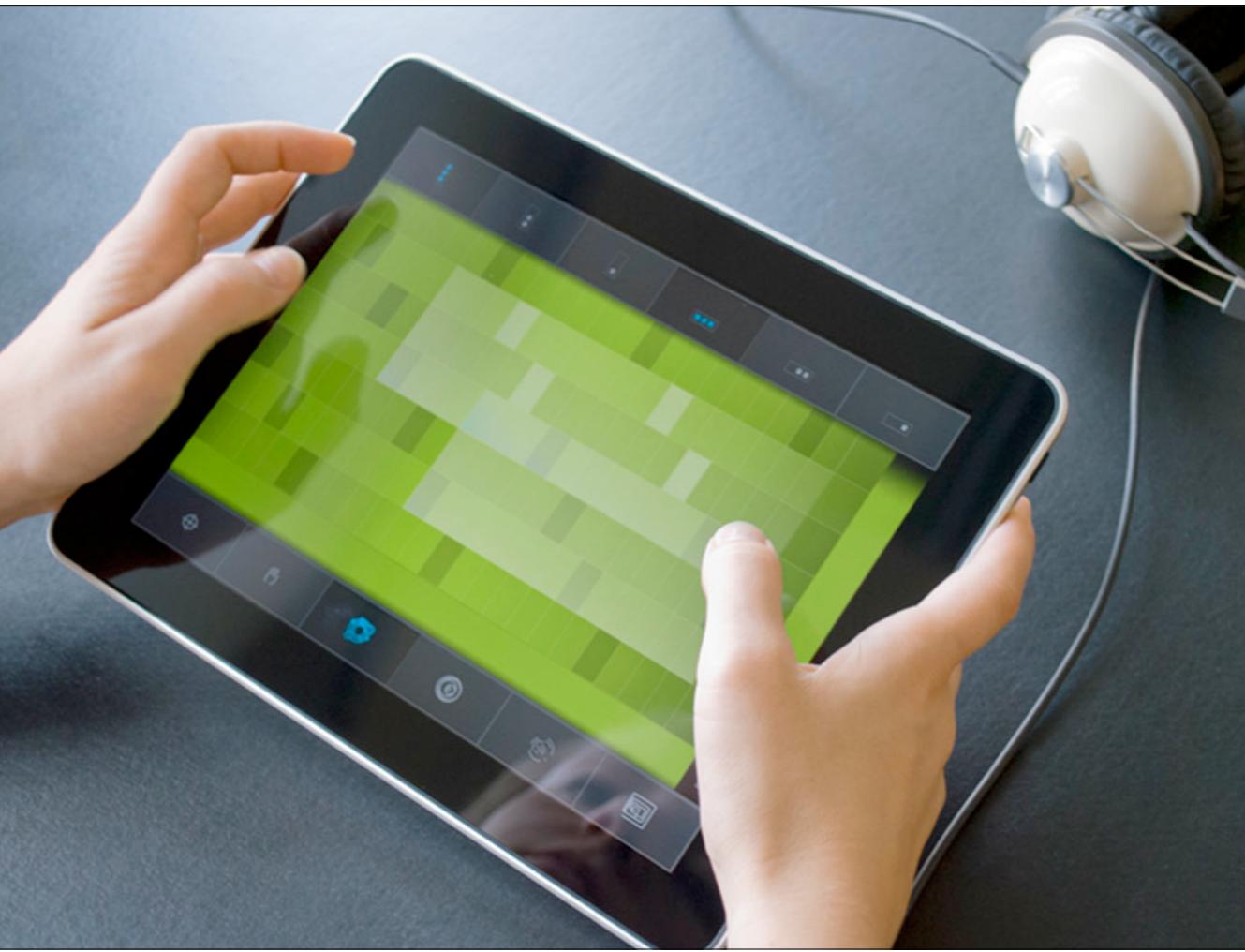


* Rekursion, integration

* Wiederverwendung, library



- * Relatives mixing
- * Dr. Jetter: "Verdammter guter Gedanke! :) Ein ewiges Ärgernis würde damit wegfallen. [...]"
- * Viele der Requirements sind so wie relatives Mixing absolute Alleinstellungsmerkmale!
- * Dr. Jetter: "Es ist völlig in Ordnung, wenn Du Dich nur auf ausgewählte Aspekte der 36 Requirements konzentrierst, schließlich beschreibst Du mit den 36 ein umfassendes System, das quasi marktreif ist."



MARKTLÜCKE

viele der requirements stellen schon einzeln ein alleinstellungsmerkmal

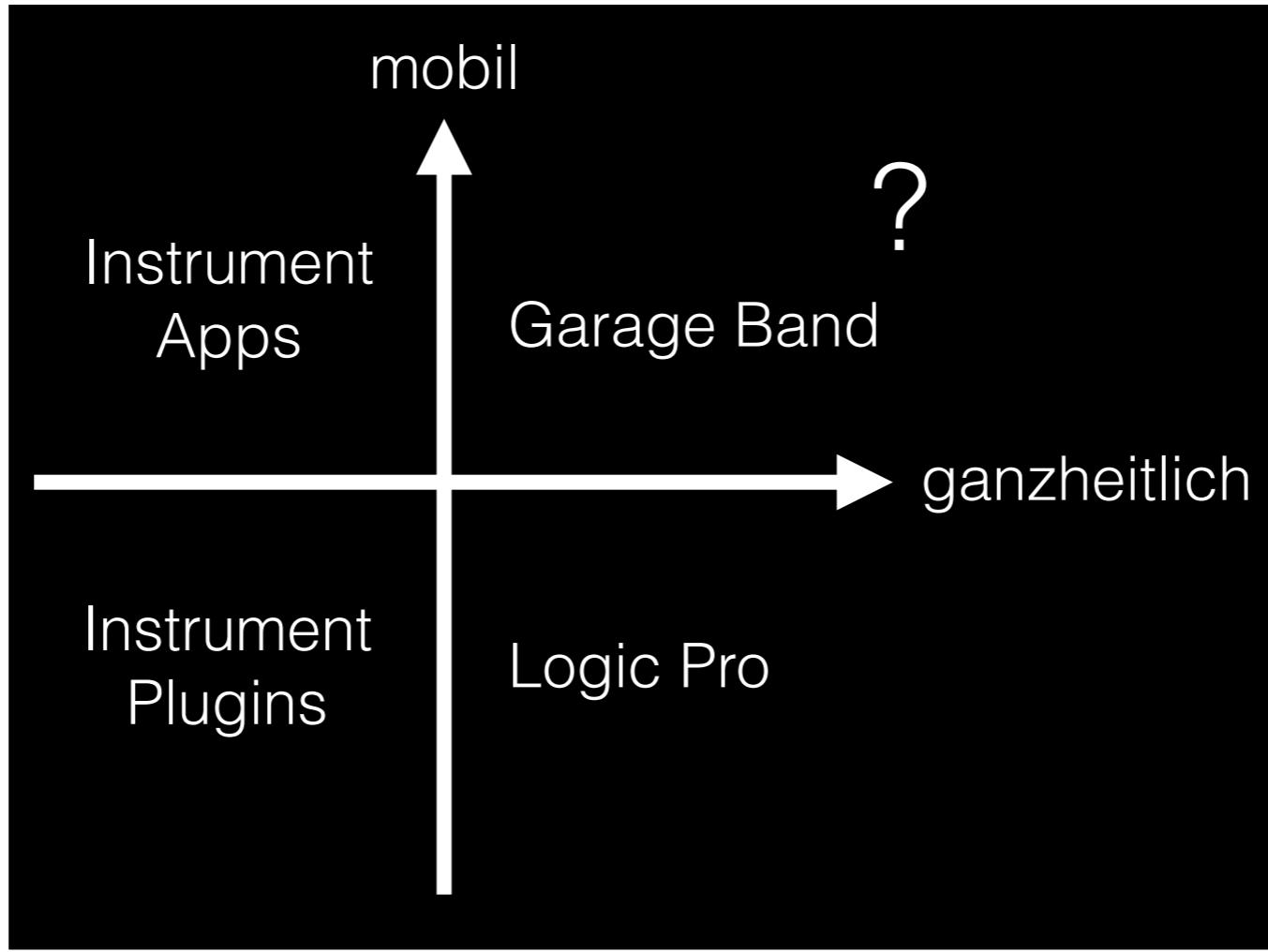
entwicklung bei mobilen Geräten

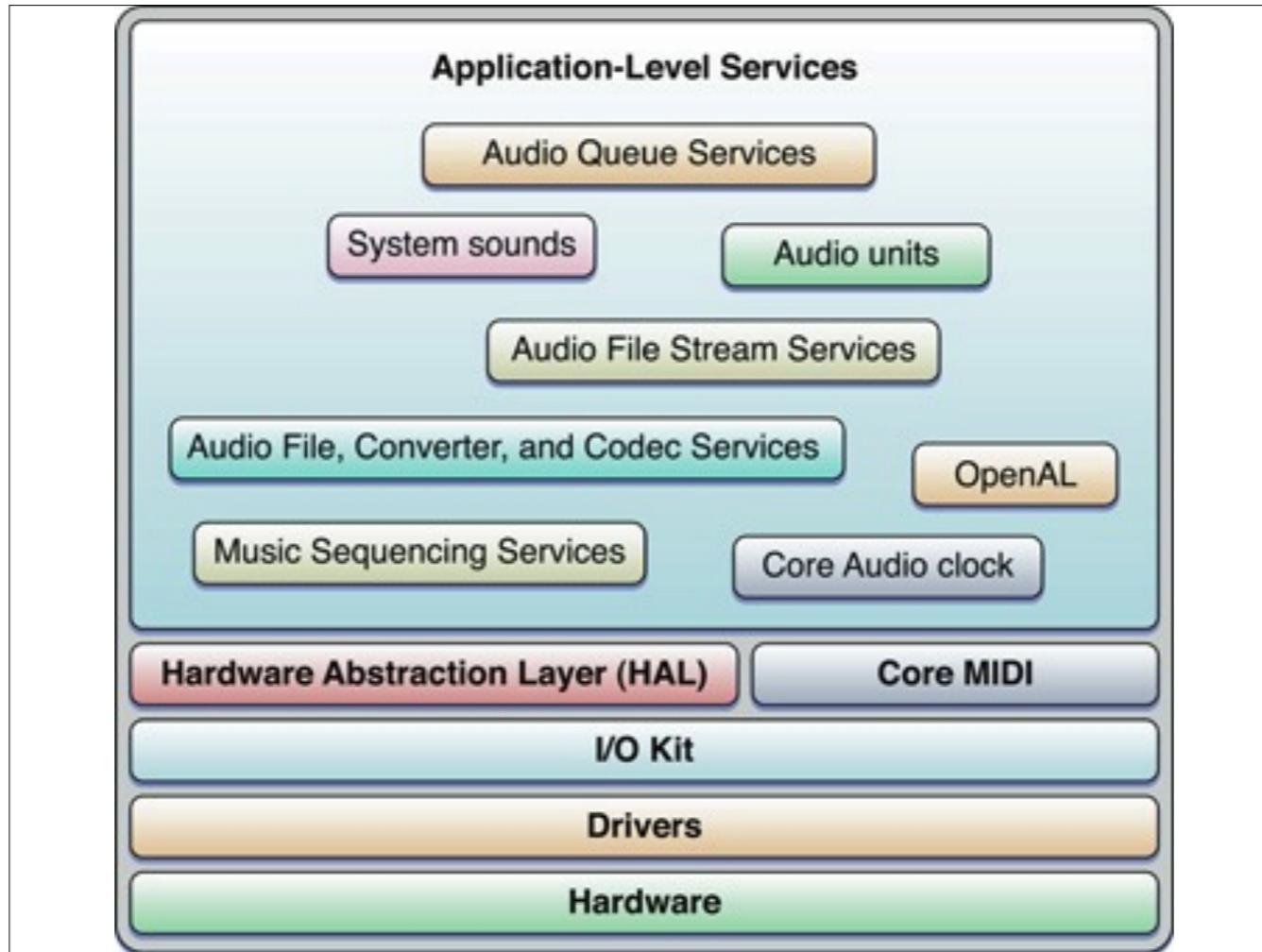
ernstzunehmende apps

mobile content production im allgemeinen

mobile audiokomposition im speziellen

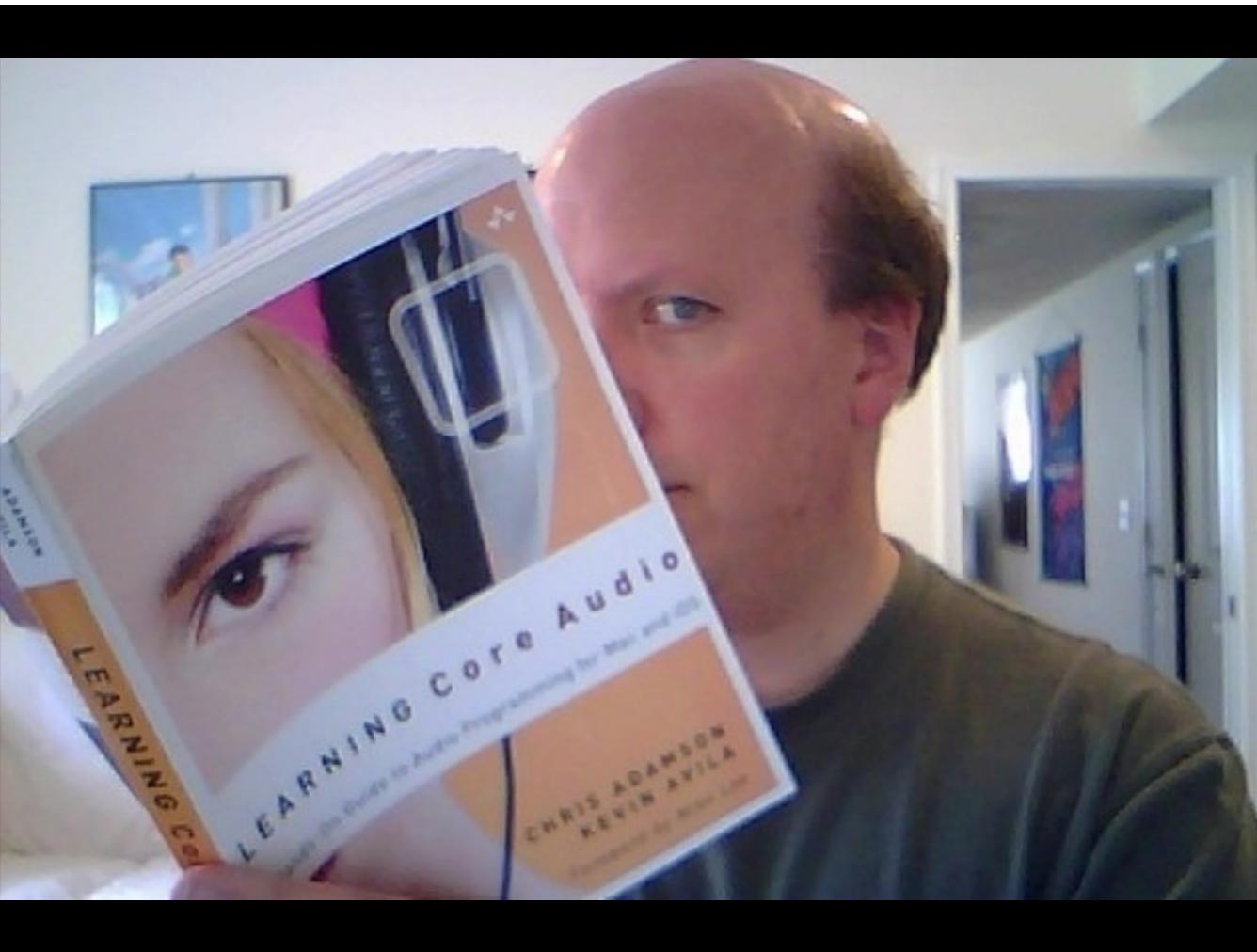
Creative (Music) games



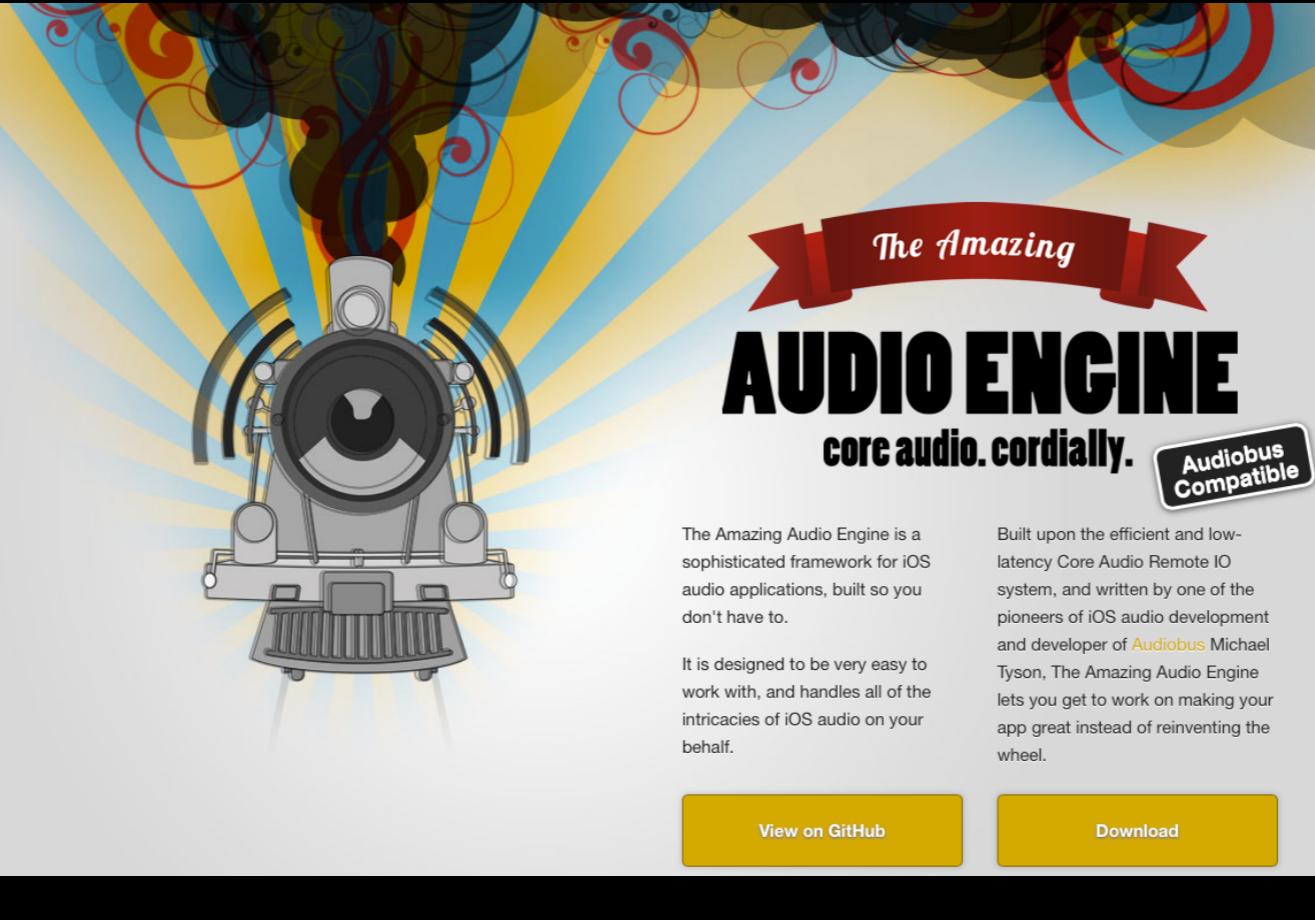


UMSETZBARKEIT

Coreaudio



Chris Adamson



The Amazing

AUDIO ENGINE

core audio. cordially.

Audiobus Compatible

The Amazing Audio Engine is a sophisticated framework for iOS audio applications, built so you don't have to. It is designed to be very easy to work with, and handles all of the intricacies of iOS audio on your behalf.

Built upon the efficient and low-latency Core Audio Remote IO system, and written by one of the pioneers of iOS audio development and developer of [Audiobus](#) Michael Tyson, The Amazing Audio Engine lets you get to work on making your app great instead of reinventing the wheel.

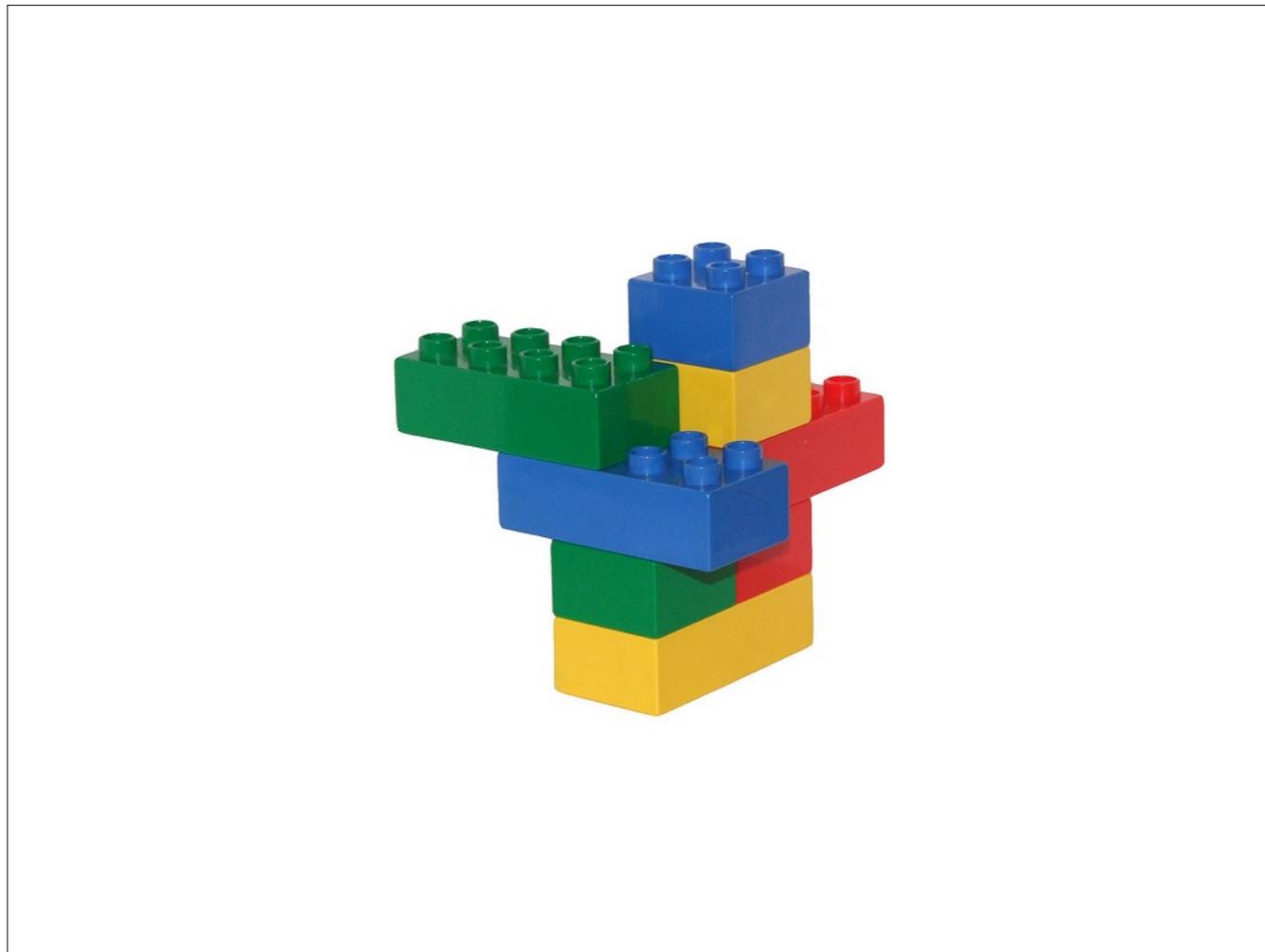
[View on GitHub](#)

[Download](#)

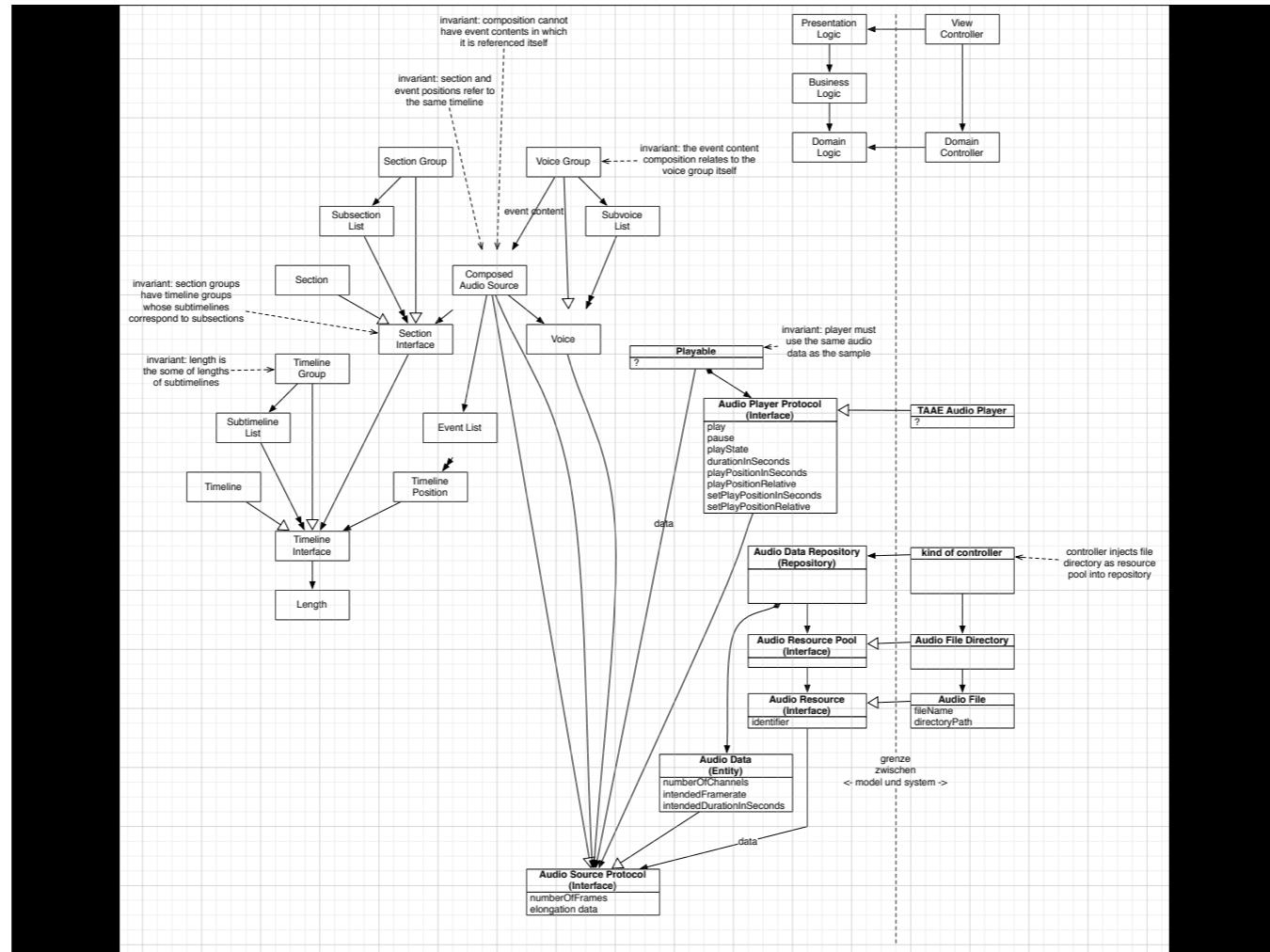
Taae

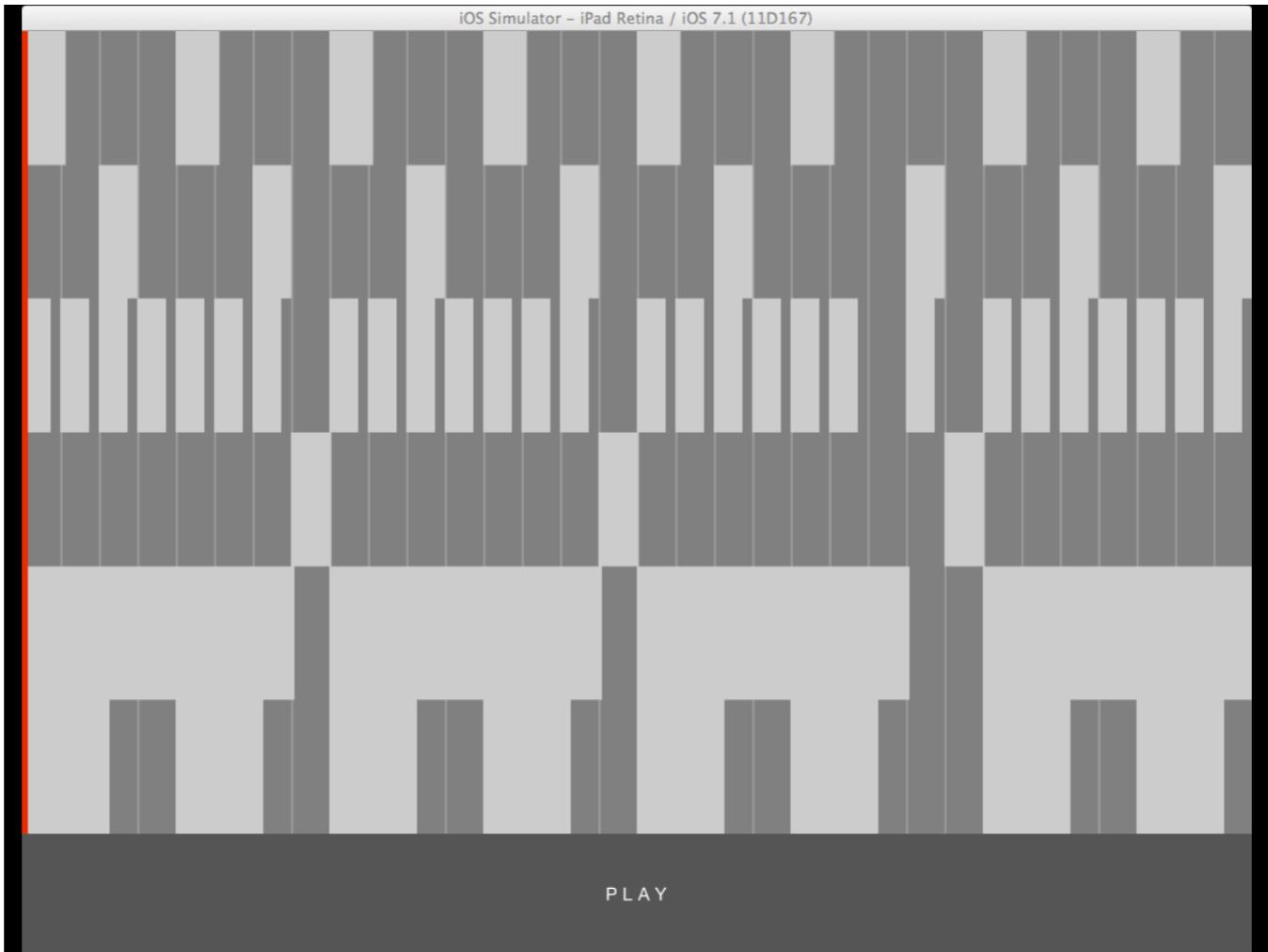


Audiobus



- * Iteratives vorgehen, Skalierbarkeit
- * Anfangs kann der User mit wenigen Konzepten schon viel experimentieren (Lego-Prinzip) -> Viel Spaß für den User bei relativ wenig Aufwand für den entwickler













VISION

- * kollaboratives komponieren
- * Competition (stackoverflow, Gamification, fame)
- * Sharing (content library in the cloud, Content Streaming)
- * licensing (user verkaufen ihren content, wir könnten mitverdienen, selbst vooxly macht das ...)

- Teamwork
- Backend
- Know How
- Marketing

WIN WIN

Was könnte formigas beitragen

- Bindung meiner Person
- Bindung der Produktidee
- Ich bin schneller 100% verfügbar
- Eigenentwicklung bei geringem Risiko
- Kommerzielles Potential

was hätte formigas von einer master-stelle?



DANKE!
hailbringer.com

INTRO

Vorstellen

Ziel des vortrages -> Vertragsvarianten

Vortrags-Zeit: 15 min

Persönlicher Hintergrund, Musikwissen

Thema: Partitur, nicht Instrument!