V2 Work Relationships Specification
DRAFT – Last updated 6/1/05 Jenn Riley

Musical works are complex entities. Musical material is heavily re-used in arrangements, transcriptions, and free interpretations of existing content. Small parts of larger works are frequently performed or published alone. Searchers for music are frequently unaware of relationships that exist, and recordings or scores of one work may meet a user's need who first searches for a different work. Two types of relationships, the derivative relationship and the whole-part relationship, identified in research into bibliographic and work relationships are particularly relevant to musical works, yet are not exploited to their fullest extent in Variations2. The following list of functional requirements outlines a proposal for how Variations2 might better document and use these two types of relationships in order to better meet user search and browse needs.

Derivative relationships

- Basic definition
    - Relationship between one source work and one derivative work based in some way on the source work. Derivative relationships include arrangements, versions, medleys, and free interpretations of source works.
- Non-exclusivity
    - A work with a source or derivative relationship to other works must also be able to have either a parent or child relationship to another work.
- Reciprocal
    - The relationship between a source and derivative work should be fully reciprocal. After it is created once, the system should make information available about the relationship from both the source and derivative work records.
    - When a relationship is deleted, it should be removed from both the source work and the derivative work.
    - When one work involved in a relationship is deleted, the relationship itself should be deleted.
- Arbitrary depth
    - Any work, even if it itself is a derivative of another work, can be the source for another derivative work.
    - Any work may have any number of derivative works.
    - Any work may be derivative of any number of source works.
- Creating the relationship
    - From any work record, a mechanism should be available to create a derivative work with the currently selected work as the source.
        - The newly-created derivative work should inherit all data elements from the source work, including the work structure. Other System IDs are an exception and should not be inherited.
        - All inherited fields in the newly-created derivative work should be editable by the cataloger.

  - A mechanism should be available to create a source-derivative relationship between two existing works.
    - A mechanism should exist for the cataloger to copy one or more selected values from the any field in the existing source work record to the existing derivative work record.
- Managing the relationship
  - A mechanism should be available to edit either end (source or derivative work) of the relationship, changing the target work of either, without requiring the cataloger to delete the relationship and create a new one.
  - From any work record, a mechanism should be available to list and open work records for derivatives and sources of the current work.
  - A mechanism should be available to reverse the direction of an existing relationship, changing the source work to the derivative and the derivative to the source.
  - Loops where a work has another work as both a source and a derivative at some hierarchical level should be prevented by the system.
- Search matches
  - When all derivatives of a source work match a query, the source work should be considered the match to the query, not its derivatives.
  - When one or some, but not all, derivatives of a source work match a query, the derivatives in question should be considered the match to the query.
- Results display
  - When a derivative work is a match to a user query, the source work should be displayed together with the derivative work. A mechanism should be available to view Containers with the source work.
  - When a source work is a match to a user query, derivatives of that source should not be displayed by default.
  - A mechanism should be available for the end-user to expand a search result set to show all derivative works of source works originally retrieved.
  - If a search match was made to a variant title, the variant title for the work should display in the search results.

Whole-part (parent-child) relationships

Note: The terms whole-part are used here interchangeably with the terms parent-child, despite the fact that these two sets of labels may have distinct meanings in other contexts.

- Basic definition
  - Relationship between one parent work and one child work representing some identifiable part of the parent work.
  - Child works are complete Variations2 works in their own right, with all the properties of works in the system.
  - Child works must always appear as work structure nodes of a parent work.
  - Work structure nodes are not necessarily all child works.
- Non-exclusivity

  - A work with a parent or child relationship to other works must also be able to have either a source or derivative relationship to another work.
- Reciprocality
  - The relationship between a parent and child work should be fully reciprocal. After it is created once, the system should make information available about the relationship from both the parent and child work records.
  - When a relationship is deleted, it should be removed from both the parent work and the child work.
  - When one work involved in a relationship is deleted, the relationship itself should be deleted.
- Arbitrary depth
  - Any work, even if it itself is a child of a parent work, can have any number of its own children.
  - Any work may have any number of child works.
  - A work may be a child of only one parent work.
- Creating the relationship
  - From any work record, a mechanism should be available to create child works from a node in the work structure of the parent work
    - A single node, any arbitrary combination of nodes, or all nodes should be selectable for creation as child works at one time.
    - The newly-created child works should inherit all data elements from the parent work. The Work Structure and Other System IDs are exceptions and should not be inherited.
    - All inherited fields in the newly-created child works should be editable by the cataloger.
  - From any work record, a mechanism should be available to create a new child work with the currently selected work as the parent.
    - The newly-created child work should inherit all data elements from the parent work. The Work Structure and Other System IDs are exceptions and should not be inherited.
    - All inherited fields in the newly-created child work should be editable by the cataloger.
    - Following the creation of the child work, a mechanism must exist to place the child work within the work structure of the parent.
  - A mechanism should be available to create a parent-child relationship between two existing works.
    - A mechanism should exist for the cataloger to copy values from the any field in the existing parent Work record to the existing child Work record, without overwriting data in fields previously populated in the child Work.
- Managing the relationship
  - A mechanism should be available to edit either end (parent or child work) of the relationship, changing the target Work of either, without requiring the cataloger to delete the relationship and create a new one.

Comment [ML1]: This section was necessary before we decided a child *must* be a work structure node. Now it might not be necessary. Do we want to have to make the structure first, then the children? If so, then we should remove this section. If we want flexibility to do either one first, we should leave this in. What do you think?

  - From any work record, a mechanism should be available to list and open work records for parents and children of the current work.
  - A mechanism should be available to reverse the direction of an existing relationship, changing the parent work to the child and the child to the parent.
  - Loops where a work has another work as both a parent and a child at some hierarchical level should be prevented by the system.
- Work structures
  - In the work structure display (in both the admin and end-user interfaces), all parents at any level of hierarchy of the currently viewed work should be displayed, with an indication of the parent-child relationship to the current work, and an indication of which is the currently-viewed work.
  - In the work structure display (in both the admin and end-user interfaces), the immediate children of a work should be displayed within the work structure, with an indication of the parent-child relationship to the current work, and an indication of which is the currently-viewed work.
  - A mechanism should exist to expand each of the child works displayed by default to view their children and work structures at any level of hierarchy, with an indication of the parent-child relationship to the current work, and an indication of which is the currently-viewed work.
  - Work structures for a given work should be viewable, creatable, and editable from a work, *and from any work that is its parent*. Data viewed in these two sources must always remain in sync.
  - When a work structure node is changed in a work with a parent relationship to the work represented by the work structure node, a mechanism should exist to ask the cataloger if the changes should be copied to the child work.
  - In the work structure display, there should be a visual indication of which work structure nodes are themselves works.
- Organizational work structure nodes
  - Nodes in a work structure should have the potential to be marked as organizational nodes, rather than content nodes. These nodes will be used to group together child works into meaningful units but will not themselves be works.
  - Organizational nodes should be excluded from creation as child works, both manually and when the command to create child works from all work structure nodes is issued.
  - Organizational nodes should not be considered parent works to work structure nodes beneath them.
  - Newly-created nodes should be content nodes (rather than organizational nodes) by default.
- Bindings
  - The work structure shown in the interface to create and edit structural bindings should include all child works and their work structures. Any child work or structural node of a child work should be allowed to be a binding point.

  - Organizational work structure nodes should be prevented from being bound to the container.

Comment [ML2]: Is this what we want?

- Titles in data model
  - A new field should be added to the Work record for a brief version of the work title, called Brief Title, as opposed to the full Uniform Title. This field should be distinct from other Variant Titles.
  - The Brief Title field will be used for results display purposes, described below, to reduce redundancy of information displayed.
- Search matches
  - Matching of work structure node labels to a query should be discontinued.
  - When all children of a parent work match a query, the parent work should be considered the match to the query, not its children.
  - When one or some, but not all, children of a parent work match a query, the children in question should be considered the match to the query.
- Results display
  - Results screens should display the work considered to be the match to the query, together with its immediate parent work, should one exist.
    - The Uniform Title should display for the parent work and all works displayed without children.
    - The Brief Title should display for the child work.
  - Results screens should not display child works of parents in search results if the children are not considered to be matches to the query.
  - A mechanism should be available for the end-user to expand a search result set to show all child works of parent works originally retrieved.
  - A mechanism should be available for the end-user to hide any child works displayed beneath parent works in a result set.
  - If a search match was made to a variant title, the variant title for the work should display in the search results.

Appendix A: Examples of derivative relationships

Explanation of the examples

Example 1: F. Schubert “Der Wanderer” D. 489 → Three versions exist. We do not really know which came first – it is possible the composer created all at the same time. These differ in key, the number of measures, performance indications, and publication date.

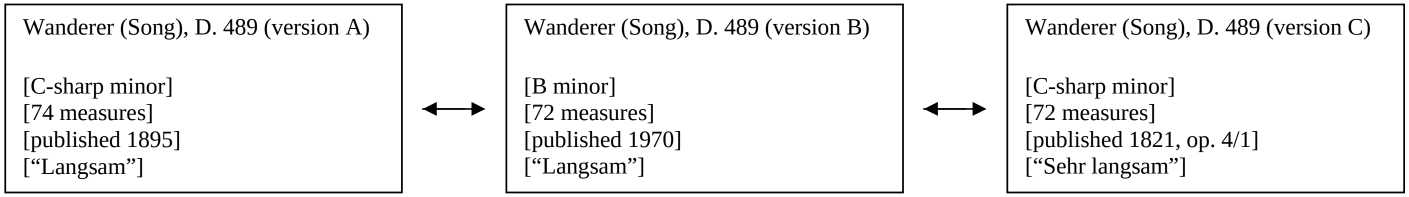
Example 2: Charles Ives “The pond” → Ives himself set this piece in a number of ways and in different groupings over time. Let me know if the diagram is not self-explanatory

Example 3: “Skanless hip-hop from the top mega-mix” → This “medley” literally quotes recordings of the other pieces and supplements the mixture with additional rap vocals

Example 4: Bach/Gounod “Ave Maria” → Gounod’s many adaptations of the 1st prelude from Bach’s *Well Tempered Klavier*, as well as further arrangements by others. (Gounod versions based upon information in New Grove Online – this is really confusing)

Example 5: (separate attachment) Bernstein *West Side story* → Bernstein wanted to rearrange the dances as a continuous work tracing the growth of various themes. Everything was reorchestrated, material was shifted around, and additional material was added. He wanted the order of material to be based on “feel” rather than show plot. While not indicated here, the scoring of the original musical was revised for the film version – this may or may not represent yet another work record. Example 4A demonstrates how the basic material was shifted around.

Example 1



Wanderer (Song), D. 489 (version A)

[C-sharp minor]
[74 measures]
[published 1895]
[“Langsam”]

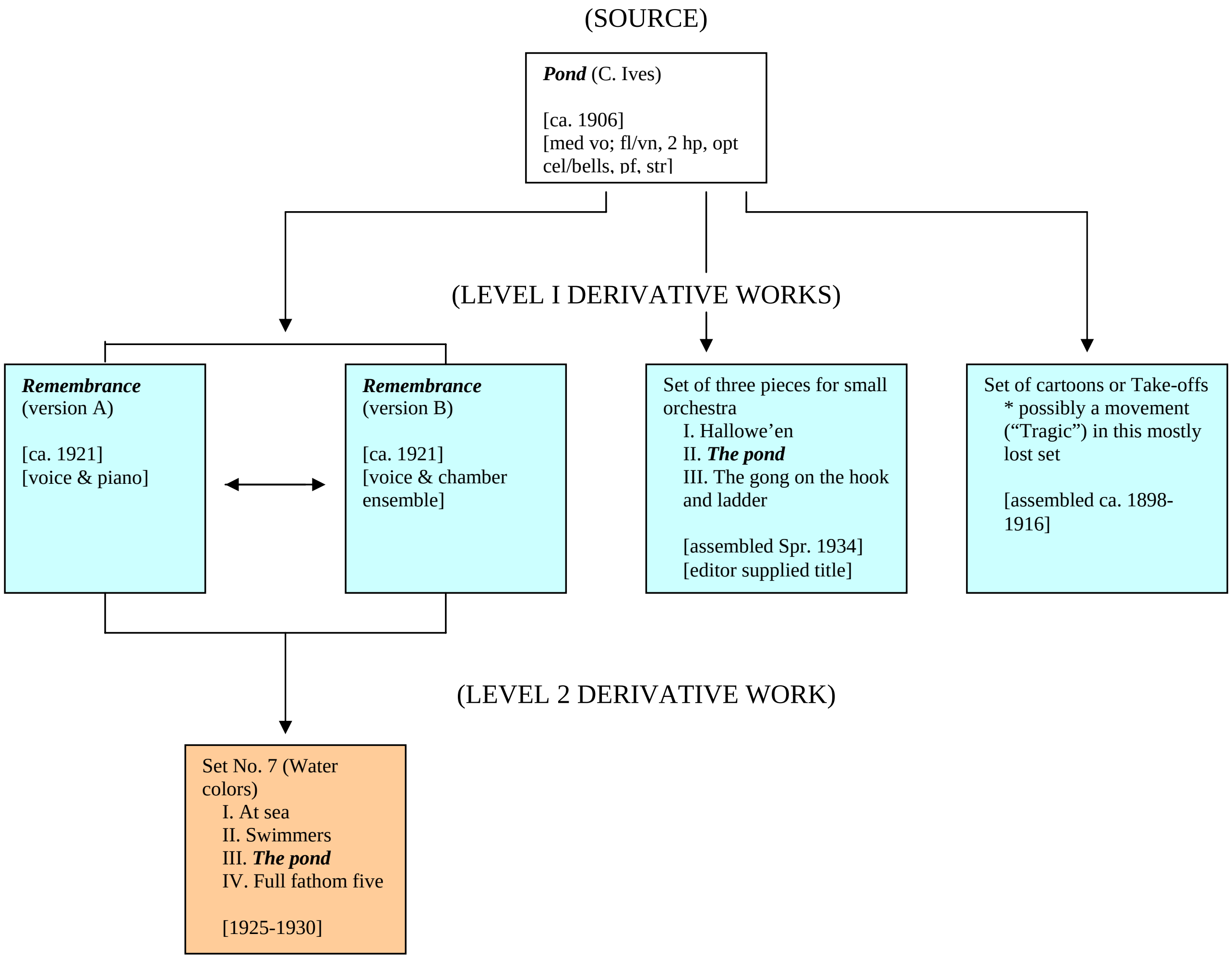
Wanderer (Song), D. 489 (version B)

[B minor]
[72 measures]
[published 1970]
[“Langsam”]

Wanderer (Song), D. 489 (version C)

[C-sharp minor]
[72 measures]
[published 1821, op. 4/1]
[“Sehr langsam”]

Example 2



(SOURCE)

Pond (C. Ives)

[ca. 1906]
[med vo; fl/vn, 2 hp, opt cel/bells, pf, str]

(LEVEL I DERIVATIVE WORKS)

Remembrance
(version A)

[ca. 1921]
[voice & piano]

Remembrance
(version B)

[ca. 1921]
[voice & chamber ensemble]

Set of three pieces for small orchestra
I. Hallowe'en
II. ***The pond***
III. The gong on the hook and ladder

[assembled Spr. 1934]
[editor supplied title]

Set of cartoons or Take-offs
* possibly a movement ("Tragic") in this mostly lost set

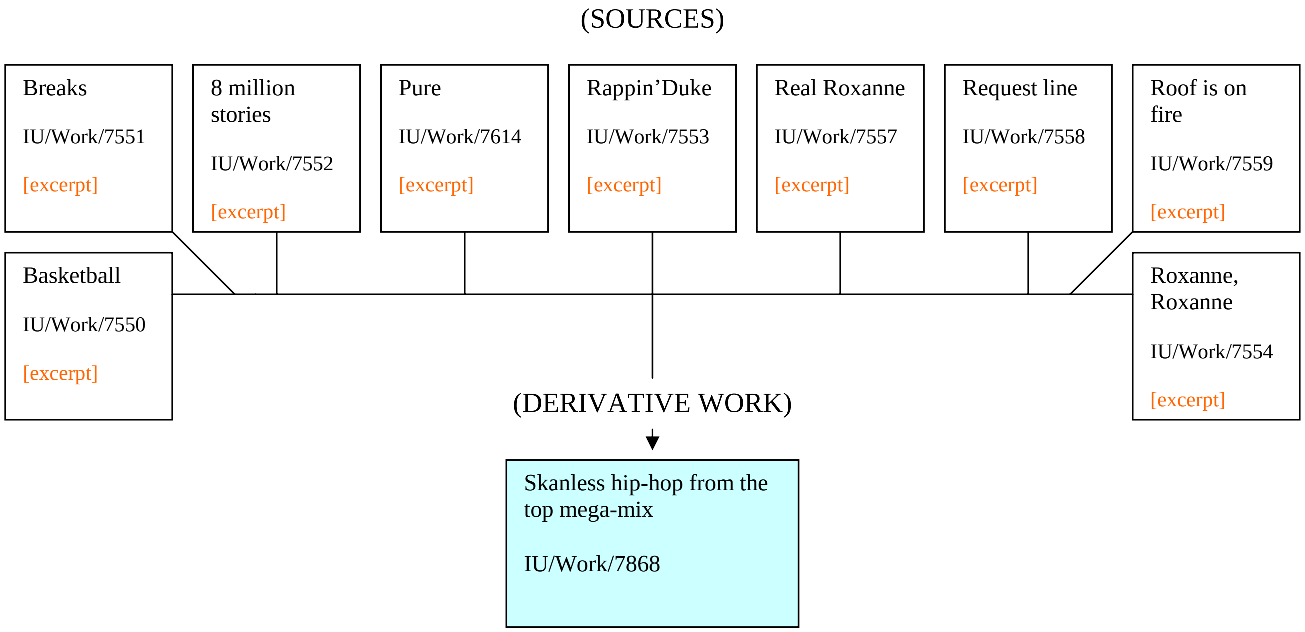
[assembled ca. 1898-1916]

(LEVEL 2 DERIVATIVE WORK)

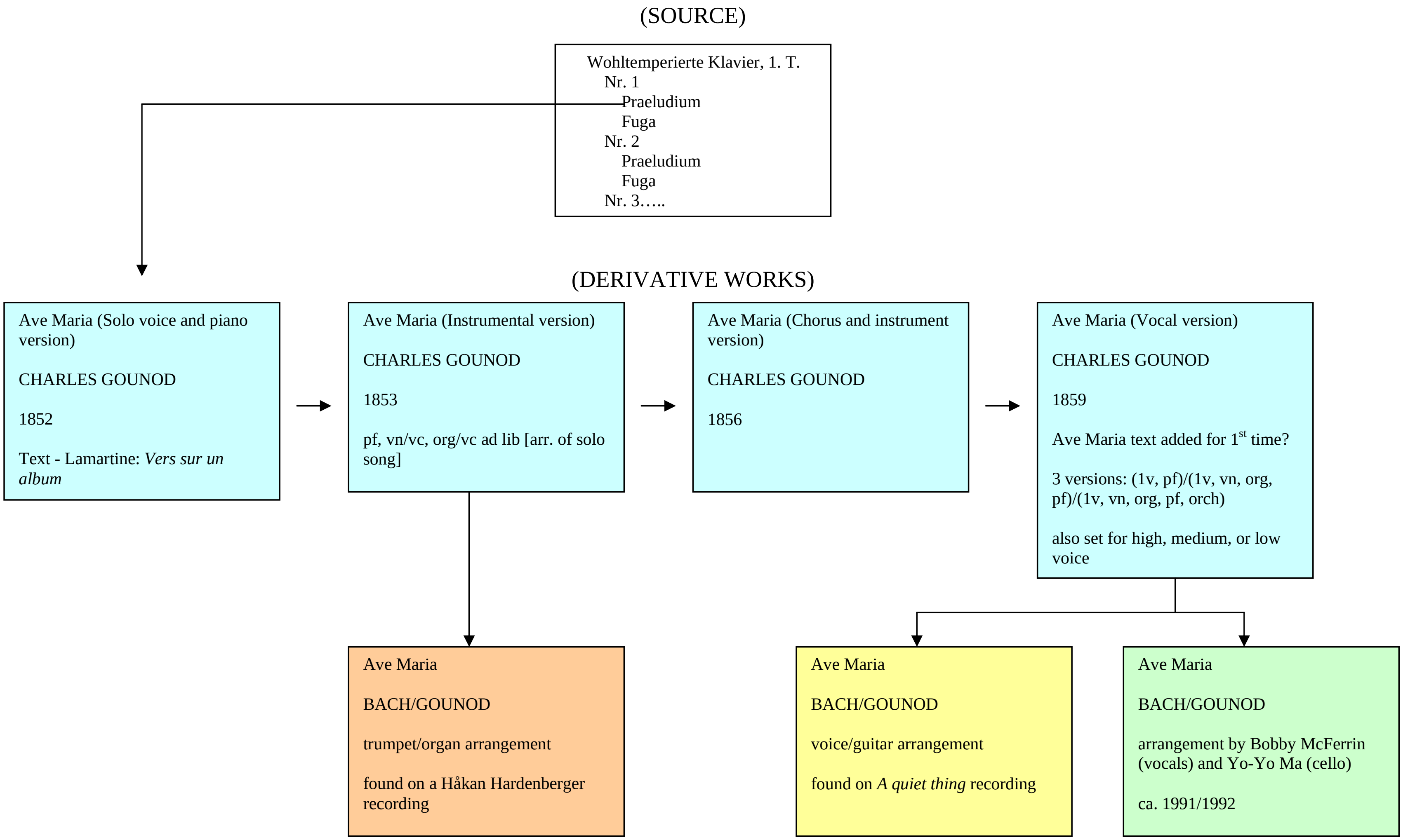
Set No. 7 (Water colors)
I. At sea
II. Swimmers
III. ***The pond***
IV. Full fathom five

[1925-1930]

Example 3


(SOURCES)
Breaks
IU/Work/7551
[excerpt]
8 million stories
IU/Work/7552
[excerpt]
Pure
IU/Work/7614
[excerpt]
Rappin'Duke
IU/Work/7553
[excerpt]
Real Roxanne
IU/Work/7557
[excerpt]
Request line
IU/Work/7558
[excerpt]
Roof is on fire
IU/Work/7559
[excerpt]
Basketball
IU/Work/7550
[excerpt]
Roxanne, Roxanne
IU/Work/7554
[excerpt]
(DERIVATIVE WORK)
Skanless hip-hop from the top mega-mix
IU/Work/7868

Example 4



(SOURCE)

Wohltemperierte Klavier, 1. T.
Nr. 1
Praeludium
Fuga
Nr. 2
Praeludium
Fuga
Nr. 3…..

(DERIVATIVE WORKS)

Ave Maria (Solo voice and piano version)

CHARLES GOUNOD

1852

Text - Lamartine: *Vers sur un album*

Ave Maria (Instrumental version)

CHARLES GOUNOD

1853

pf, vn/vc, org/vc ad lib [arr. of solo song]

Ave Maria (Chorus and instrument version)

CHARLES GOUNOD

1856

Ave Maria (Vocal version)

CHARLES GOUNOD

1859

Ave Maria text added for 1st time?

3 versions: (1v, pf)/(1v, vn, org, pf)/(1v, vn, org, pf, orch)

also set for high, medium, or low voice

Ave Maria

BACH/GOUNOD

trumpet/organ arrangement

found on a Håkan Hardenberger recording

Ave Maria

BACH/GOUNOD

voice/guitar arrangement

found on *A quiet thing* recording

Ave Maria

BACH/GOUNOD

arrangement by Bobby McFerrin (vocals) and Yo-Yo Ma (cello)

ca. 1991/1992

Example 5

(SOURCE)

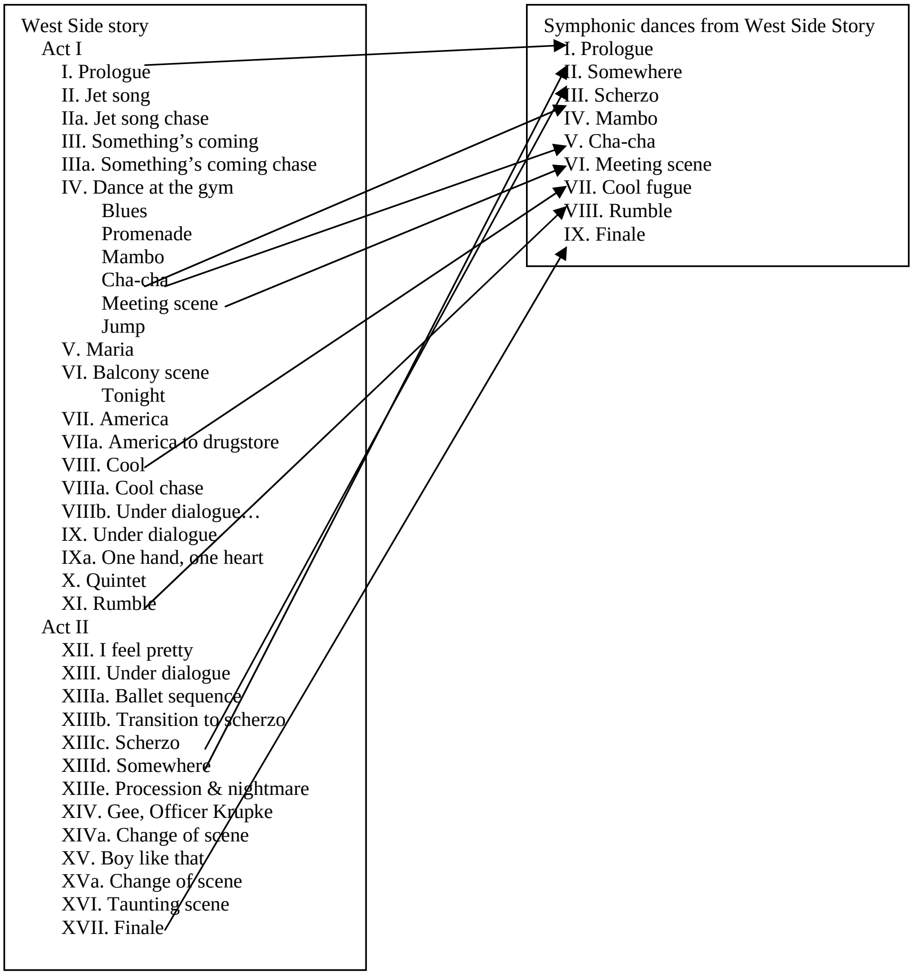
- West Side story
  - Act I
    - I. Prologue
    - II. Jet song
    - IIa. Jet song chase
    - III. Something’s coming
    - IIIa. Something’s coming chase
    - IV. Dance at the gym
      - Blues
      - Promenade
      - Mambo
      - Cha-cha
      - Meeting scene
      - Jump
    - V. Maria
    - VI. Balcony scene
      - Tonight
    - VII. America
    - VIIa. America to drugstore
    - VIII. Cool
    - VIIIa. Cool chase
    - VIIIb. Under dialogue…
    - IX. Under dialogue
    - IXa. One hand, one heart
    - X. Quintet
    - XI. Rumble
  - Act II
    - XII. I feel pretty
    - XIII. Under dialogue
    - XIIIa. Ballet sequence
    - XIIIb. Transition to scherzo
    - XIIIc. Scherzo
    - XIIId. Somewhere
    - XIIIe. Procession & nightmare
    - XIV. Gee, Officer Krupke
    - XIVa. Change of scene
    - XV. Boy like that
    - XVa. Change of scene
    - XVI. Taunting scene
    - XVII. Finale

→

(DERIVATIVE WORK)

- Symphonic dances from West Side Story
  - I. Prologue
  - II. Somewhere
  - III. Scherzo
  - IV. Mambo
  - V. Cha-cha
  - VI. Meeting scene
  - VII. Cool fugue
  - VIII. Rumble
  - IX. Finale

Example 5A



West Side story

Act I

- I. Prologue
- II. Jet song
- IIa. Jet song chase
- III. Something’s coming
- IIIa. Something’s coming chase
- IV. Dance at the gym
  - Blues
  - Promenade
  - Mambo
  - Cha-cha
  - Meeting scene
  - Jump
- V. Maria
- VI. Balcony scene
  - Tonight
- VII. America
- VIIa. America to drugstore
- VIII. Cool
- VIIIa. Cool chase
- VIIIb. Under dialogue…
- IX. Under dialogue
- IXa. One hand, one heart
- X. Quintet
- XI. Rumble

Act II

- XII. I feel pretty
- XIII. Under dialogue
- XIIIa. Ballet sequence
- XIIIb. Transition to scherzo
- XIIIc. Scherzo
- XIIId. Somewhere
- XIIIe. Procession & nightmare
- XIV. Gee, Officer Krupke
- XIVa. Change of scene
- XV. Boy like that
- XVa. Change of scene
- XVI. Taunting scene
- XVII. Finale

Symphonic dances from West Side Story

- I. Prologue
- II. Somewhere
- III. Scherzo
- IV. Mambo
- V. Cha-cha
- VI. Meeting scene
- VII. Cool fugue
- VIII. Rumble
- IX. Finale

Appendix B. Examples of whole/part relationships

Black entries = separate work records
Red entries = organizational nodes (not separate work records)
Blue entries = links to records in the upper levels of the hierarchy

Highest Level Work Record

(collapsed)

Ring des Nibelungen (R. Wagner)

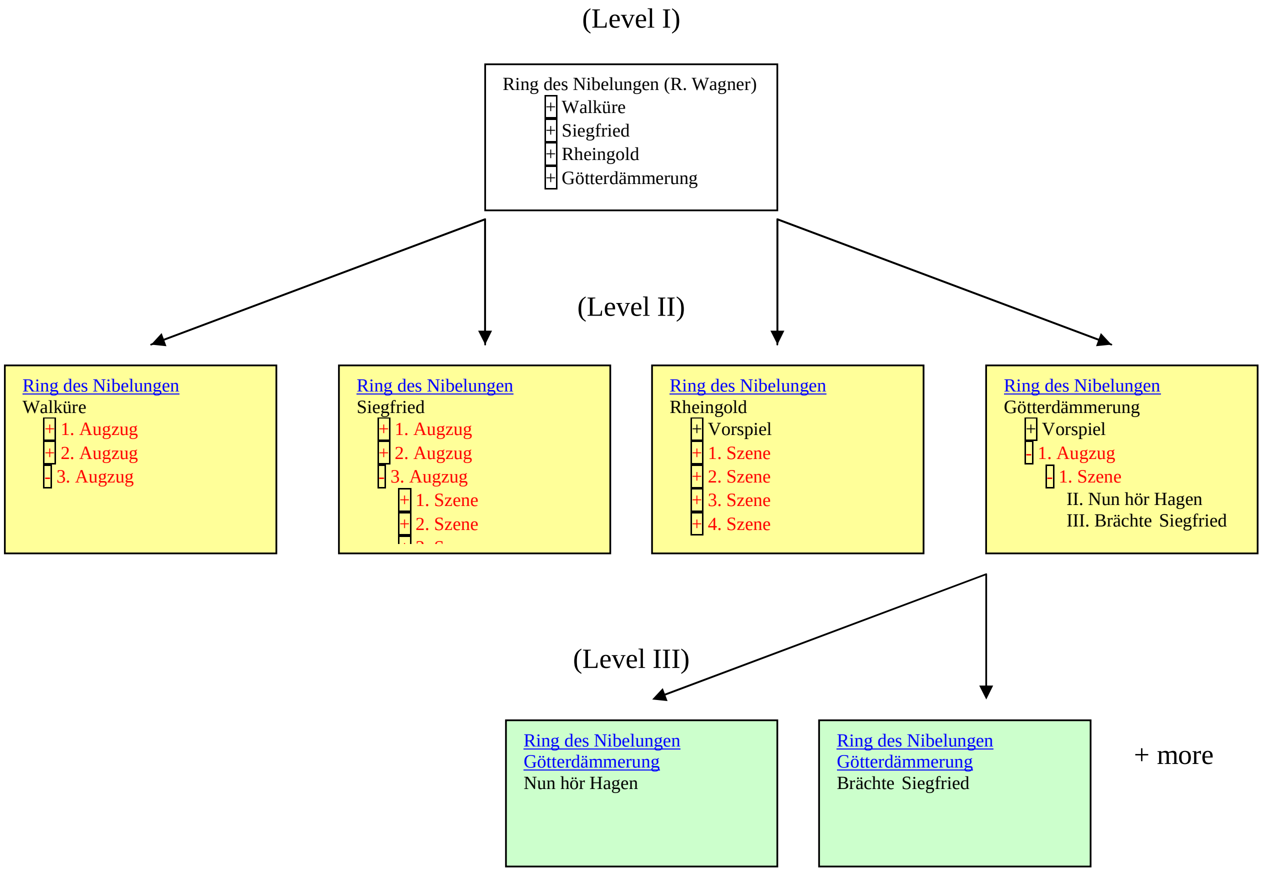
- ⊞ Walküre
- ⊞ Siegfried
- ⊞ Rheingold
- ⊞ Götterdämmerung

(partially expanded)

Ring des Nibelungen (R. Wagner)

- ⊞ Walküre
- ⊞ Siegfried
- ⊞ Rheingold
- ⊟ Götterdämmerung
  - ⊟ Vorspiel
    - I. Zu neuen Thaten
  - ⊟ 1. Augzug
    - ⊟ 1. Szene
      - II. Nun hör Hagen
      - III. Brächte Siegfried

Partial Work Record Hierarchy for Wagner's Ring Cycle



(Level I)

Ring des Nibelungen (R. Wagner)
⊞ Walküre
⊞ Siegfried
⊞ Rheingold
⊞ Götterdämmerung

(Level II)

Ring des Nibelungen
Walküre
⊞ 1. Augzug
⊞ 2. Augzug
⊟ 3. Augzug

Ring des Nibelungen
Siegfried
⊞ 1. Augzug
⊞ 2. Augzug
⊟ 3. Augzug
⊞ 1. Szene
⊞ 2. Szene

Ring des Nibelungen
Rheingold
⊞ Vorspiel
⊞ 1. Szene
⊞ 2. Szene
⊞ 3. Szene
⊞ 4. Szene

Ring des Nibelungen
Götterdämmerung
⊞ Vorspiel
⊟ 1. Augzug
⊟ 1. Szene
II. Nun hör Hagen
III. Brächte Siegfried

(Level III)

Ring des Nibelungen
Götterdämmerung
Nun hör Hagen

Ring des Nibelungen
Götterdämmerung
Brächte Siegfried

+ more