Cal Poly Bioinformatics:

Designing a Cross-Disciplinary Minor that Builds on the In-Concert Model for Bioinformatics Education

Jean Davison, Biology
Alex Dekhtyar, Computer Science
Anya Goodman, Chemistry/Biochemistry
Ed Himelblau, Biology
Franz Kurfess, Computer Science
Theresa Migler-VonDollen, Computer Science
Andrew Schaffner, Statistics

Thank you!
CAWIT Technology Pathways Initiative

Natalie Jo Schafer, Cal Poly Strategic Development Initiatives

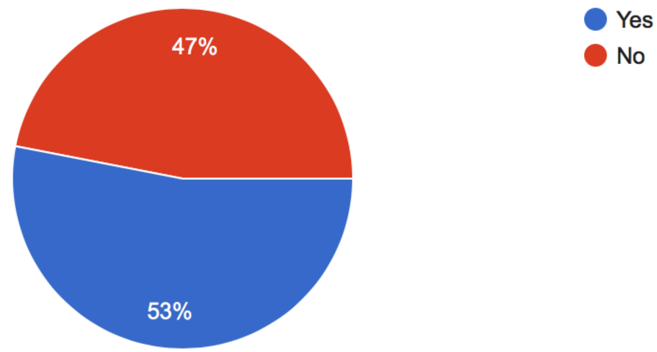
CAL POLY
SAN LUIS OBISPO

Prologue: Interest in and Challenges of Bioinformatics at Cal Poly

Interest in Bioinformatics:
(among 164 Biology Freshman)

Would you consider taking additional coursework (beyond the credits required for your degree) to help you add coding experience?

164 responses



Limited Opportunities:

BIO 441 (Bioinformatics Applications): about 40 students per year
CSC 448 (Bioinformatics Algorithms): about 40 students per year

Difficult for BIO/BCHEM Students to enroll in Computer Science courses (e.g. very few complete existing Computer Science minor due to scheduling constraints)

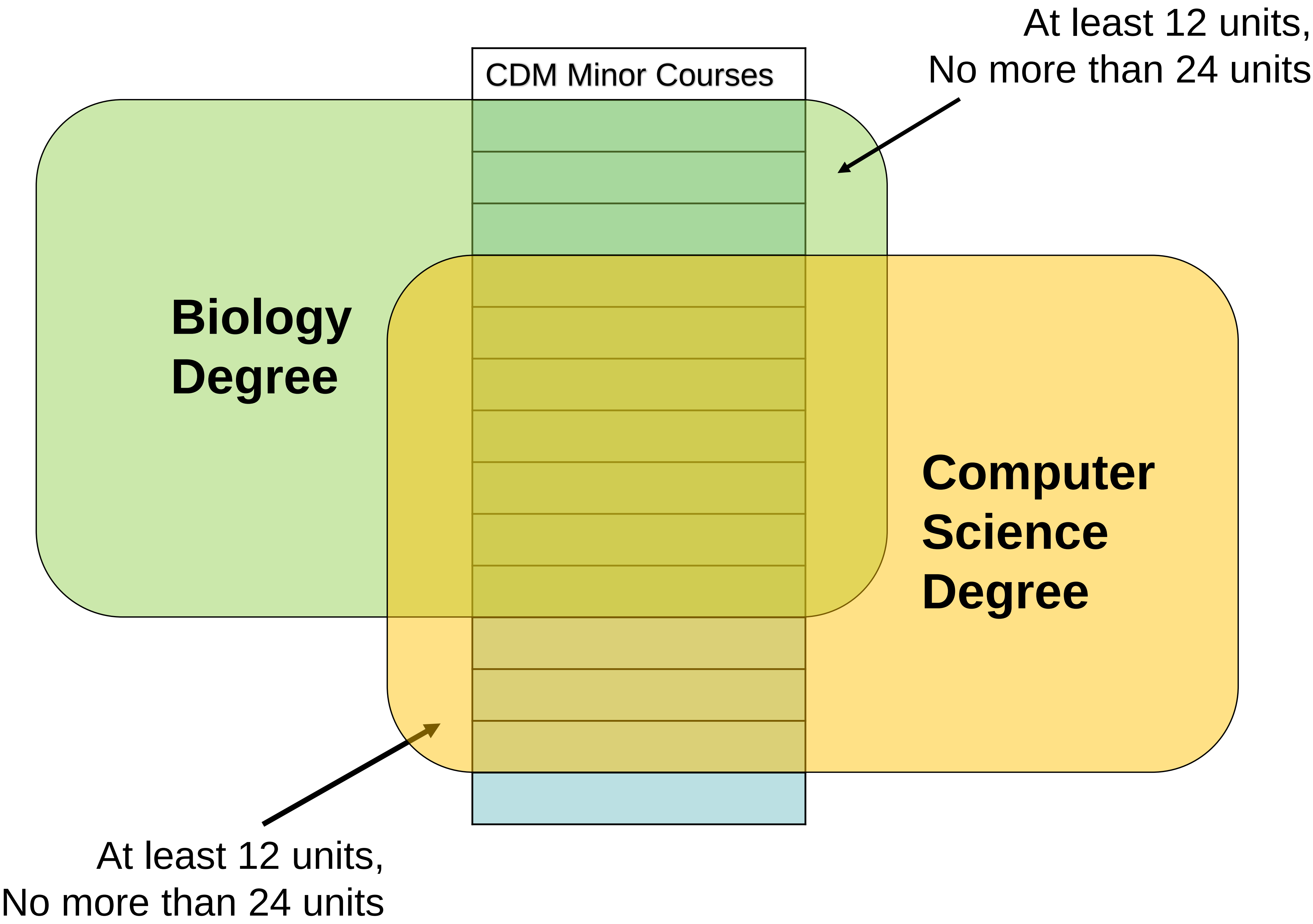
Create a Minor in Bioinformatics (Supported by CAWIT/TPI) that is informed by our experience teaching bioinformatics in a cross-disciplinary way (“in concert”).

Only open (or possible!) for particular majors.

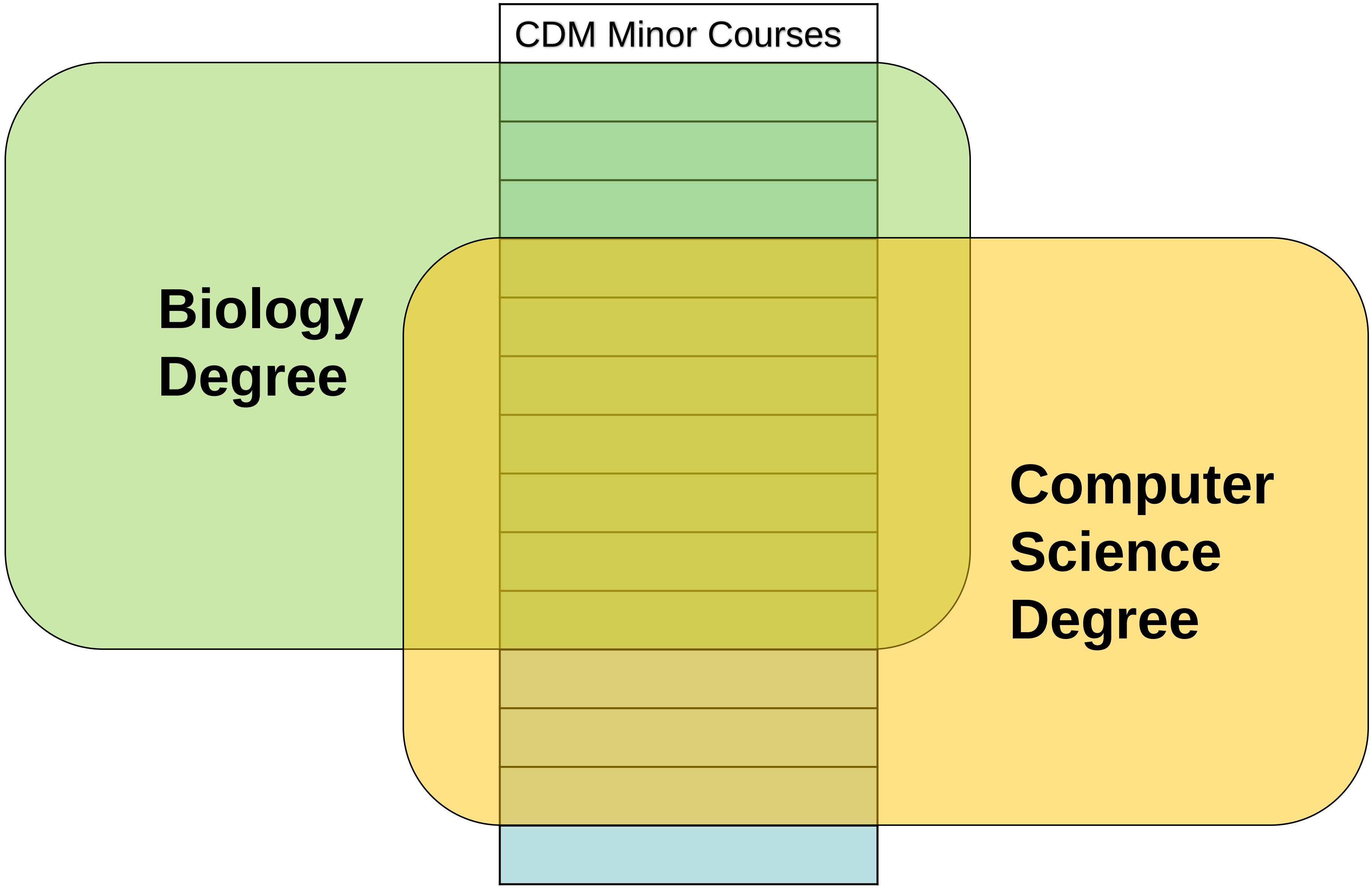
All student in the CDM take the same courses regardless of major.

At least 12 and no more than 24 units must be from outside the student’s major degree.

Cross-Disciplinary Minors at Cal Poly



Cross-Disciplinary Minors at Cal Poly



MESSAGE FROM ISCB

Bioinformatics Curriculum Guidelines: Toward a Definition of Core Competencies

Lonnie Welch, Fran Lewitter, Russell Schwartz, Cath Brooksbank, Predrag Radivojac, Bruno Gaeta, Maria Victoria Schneider

Published: March 6, 2014 • https://doi.org/10.1371/journal.pcbi.1003496

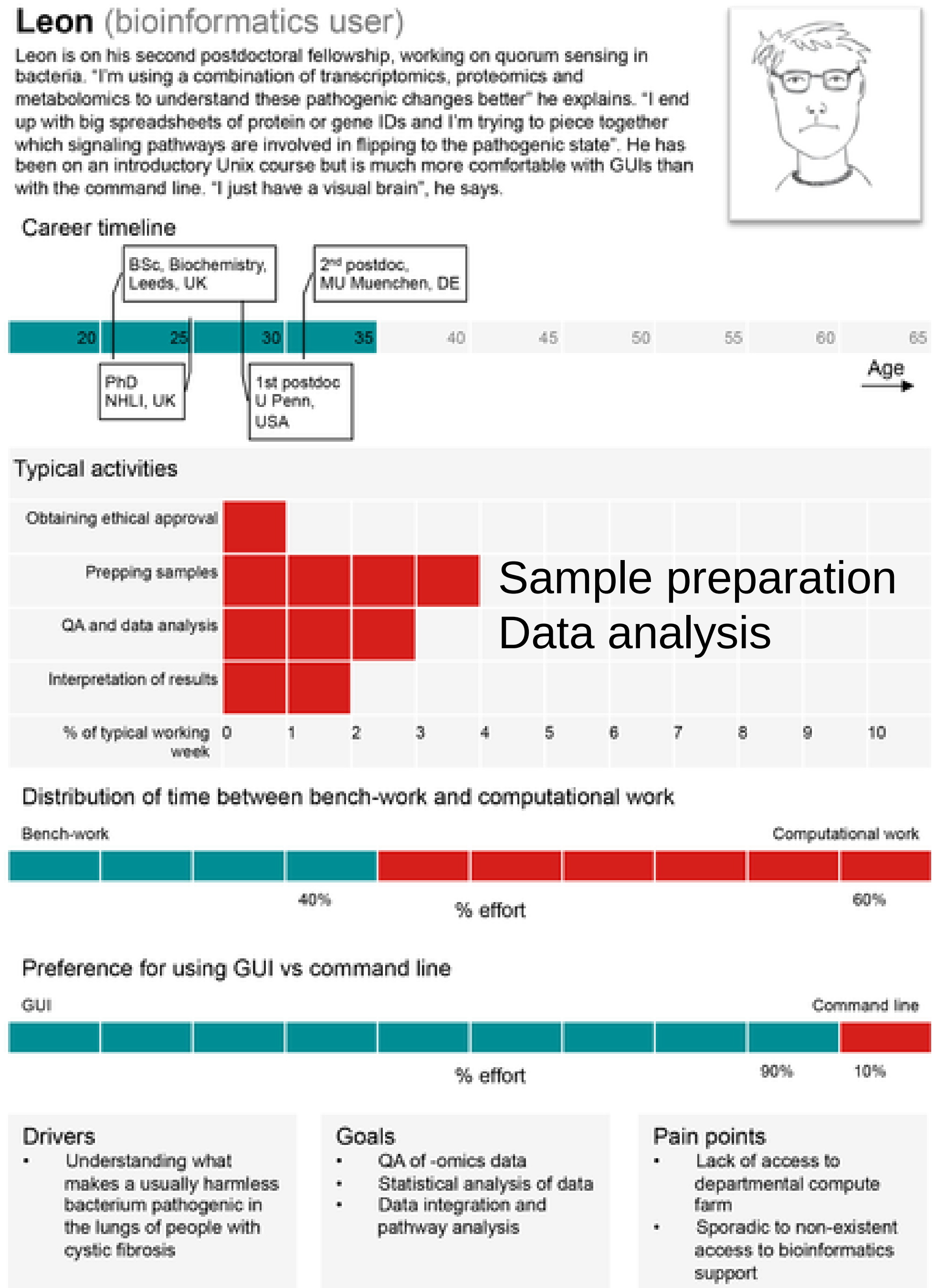
Bioinformatics User

career path

typical activities

lab bench

GUI



workstation

command Line

Welch L, Lewitter F, Schwartz R, Brooksbank C, Radivojac P, et al. (2014) Bioinformatics Curriculum Guidelines: Toward a Definition of Core Competencies. PLOS Computational Biology 10(3): e1003496. https://doi.org/10.1371/journal.pcbi.1003496
http://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1003496

Bioinformatics Scientist

career path

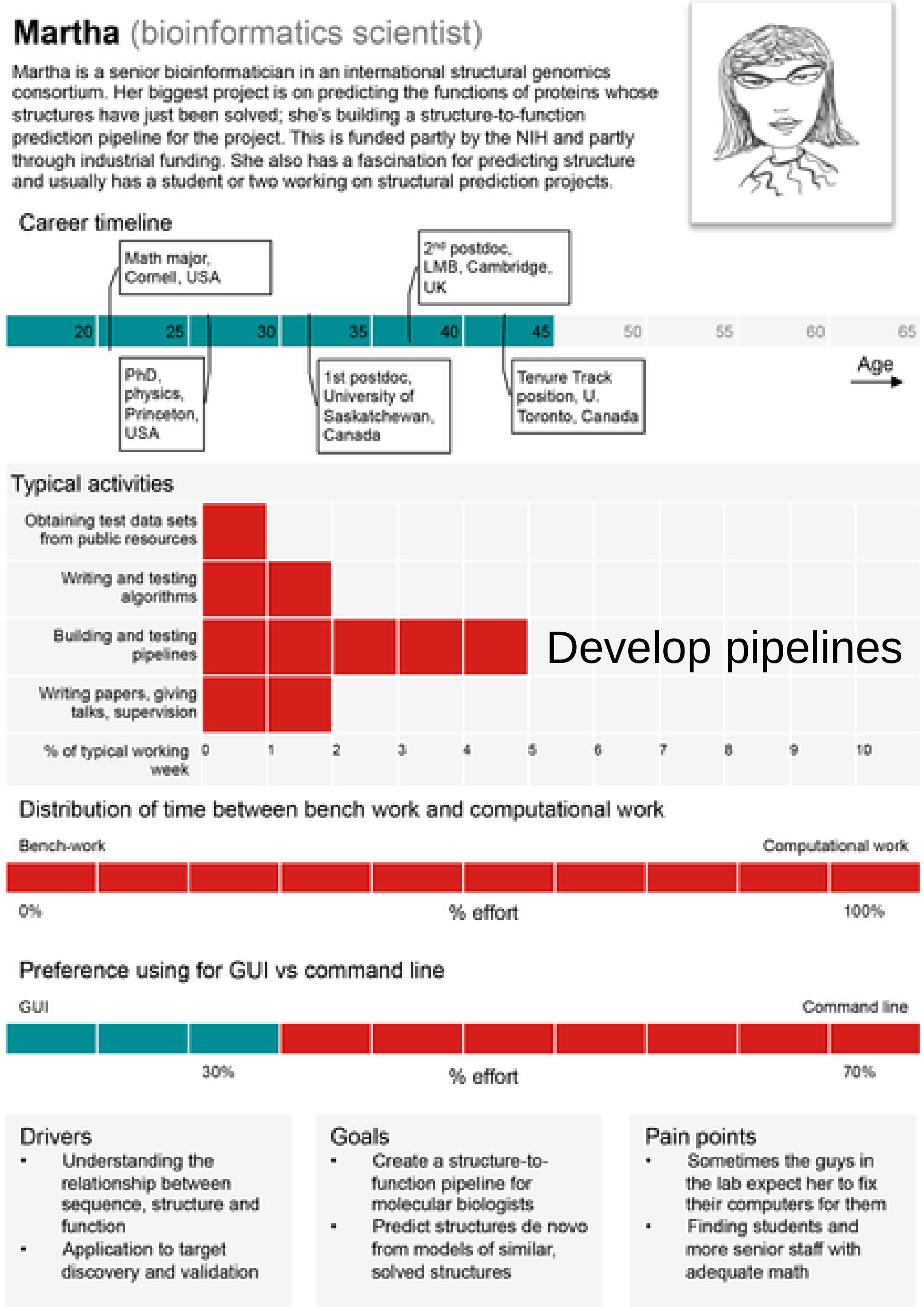
typical activities

Develop pipelines

100% workstation

GUI

command Line



Welch L, Lewitter F, Schwartz R, Brooksbank C, Radivojac P, et al. (2014) Bioinformatics Curriculum Guidelines: Toward a Definition of Core Competencies. PLOS Computational Biology 10(3): e1003496. https://doi.org/10.1371/journal.pcbi.1003496
http://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1003496

Bioinformatics Engineer

career path

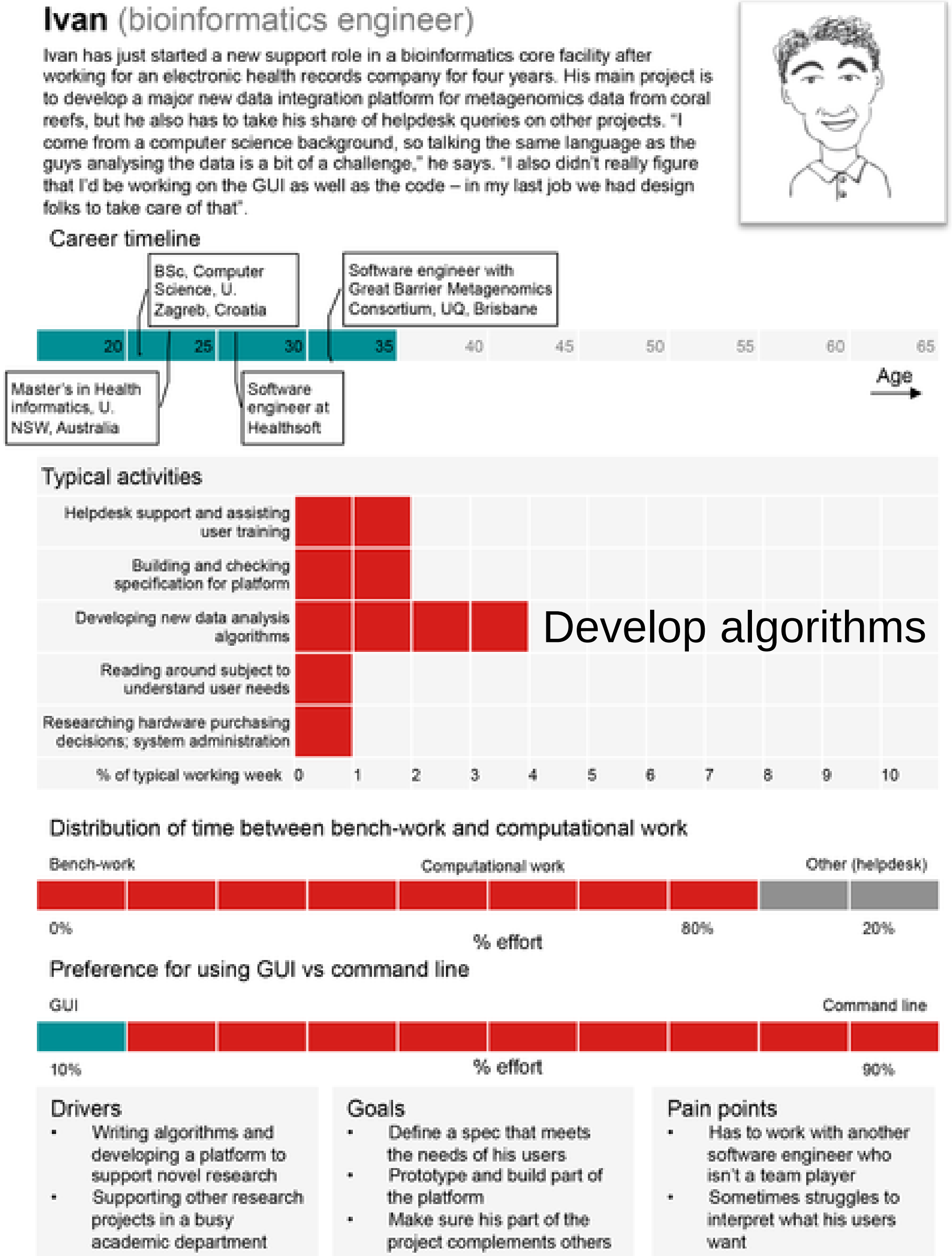
typical activities

Develop algorithms

100% workstation

GUI

command Line

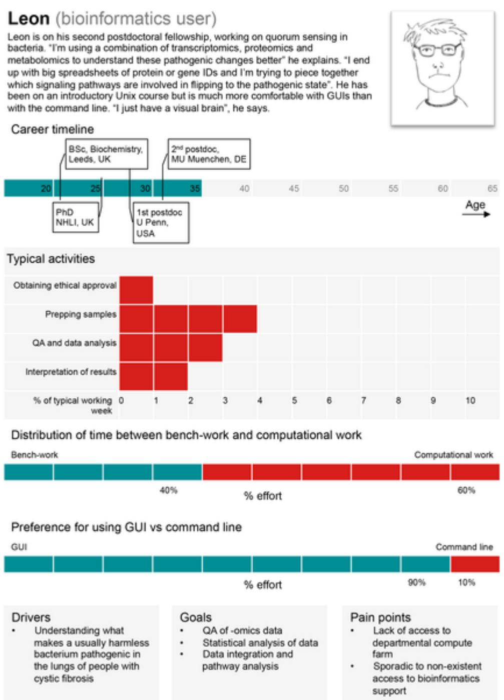


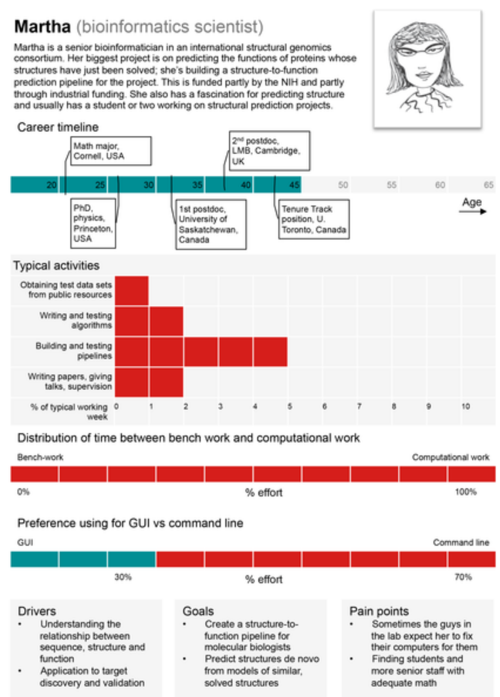
Welch L, Lewitter F, Schwartz R, Brooksbank C, Radivojac P, et al. (2014) Bioinformatics Curriculum Guidelines: Toward a Definition of Core Competencies. PLOS Computational Biology 10(3): e1003496. https://doi.org/10.1371/journal.pcbi.1003496
http://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1003496

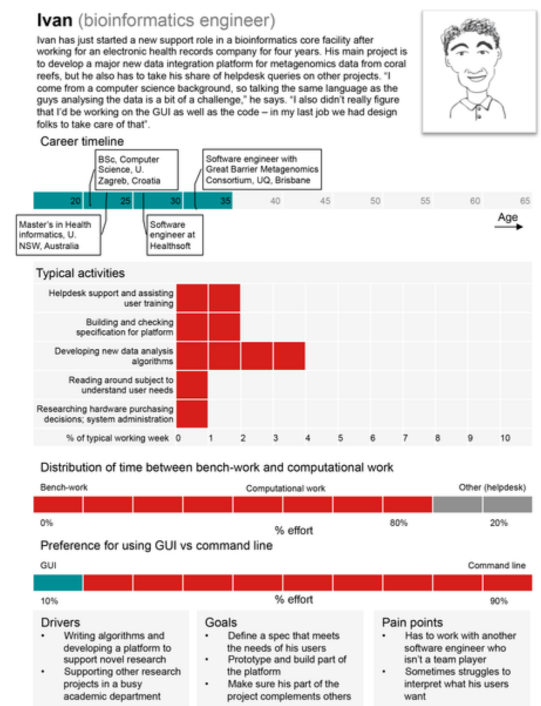
Bioinformatics Minor Degree Holder

Well prepared to apply concepts and skills from outside her primary degree area.

Proficient at cross-disciplinary collaboration in a bioinformatics setting.







Welch L, Lewitter F, Schwartz R, Brooksbank C, Radivojac P, et al. (2014) Bioinformatics Curriculum Guidelines: Toward a Definition of Core Competencies. PLOS Computational Biology 10(3): e1003496. https://doi.org/10.1371/journal.pcbi.1003496
http://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1003496

Computation

Programming

Software Engineering

Algorithm Design and Analysis

Machine learning/data mining

Databases

Scripting Languages

Ability to use stats software packages

Open source repositories

Web UI

Version Control

Core Competencies

Computation

- Programming
- Software Engineering
- Algorithm Design and Analysis
- Machine learning/data mining
- Databases
- Scripting Languages
- Ability to use stats software packages
- Open source repositories
- Web UI
- Version Control

Biology

- Molecular biology
- Genetics
- Genomics (-omics)
- Evolution

Core Competencies

Computation

Programming

Software Engineering

Algorithm Design and Analysis

Machine learning/data mining

Databases

Scripting Languages

Ability to use stats software packages

Open source repositories

Web UI

Version Control

Biology

Molecular biology

Genetics

Genomics (-omics)

Evolution

Bioinformatics

Retrieve and manipulate public data

Use (manage, interpret, analyze) large data sets

Explain common bioinformatics algorithms.

Access bioinformatics resources

Communicate genomic data

Core Competencies

Computation

Programming

Software Engineering

Algorithm Design and Analysis

Machine learning/data mining

Databases

Scripting Languages

Ability to use stats software packages

Open source repositories

Web UI

Version Control

Biology

Molecular biology

Genetics

Genomics (-omics)

Evolution

Bioinformatics

Retrieve and manipulate public data

Use (manage, interpret, analyze) large data sets

Explain common bioinformatics algorithms.

Access bioinformatics resources

Communicate genomic data

Statistics

Descriptive Statistics

Data Visualization

Experimental design

Statistical Inference

Statistical Models (intro to regression / ANOVA)

R and R libraries

Statistical Software

Probability

Core Competencies

Computation

Programming

Software Engineering

Algorithm Design and Analysis

Machine learning/data mining

Databases

Scripting Languages

Ability to use stats software packages

Open source repositories

Web UI

Version Control

Biology

Molecular biology

Genetics

Genomics (-omics)

Evolution

Bioinformatics

Retrieve and manipulate public data

Use (manage, interpret, analyze) large data sets

Explain common bioinformatics algorithms.

Access bioinformatics resources

Communicate genomic data

Statistics

Descriptive Statistics

Data Visualization

Experimental design

Statistical Inference

Statistical Models (intro to regression / ANOVA)

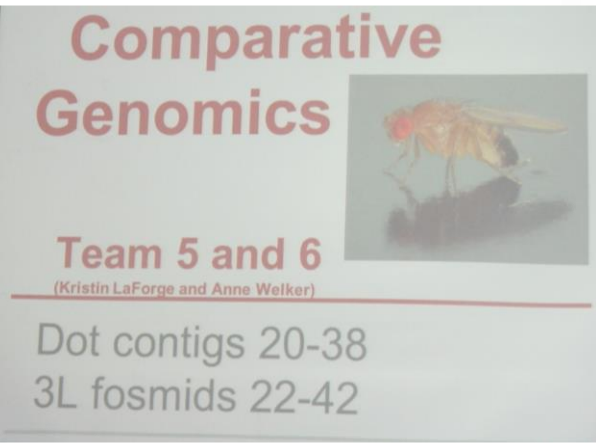
R and R libraries

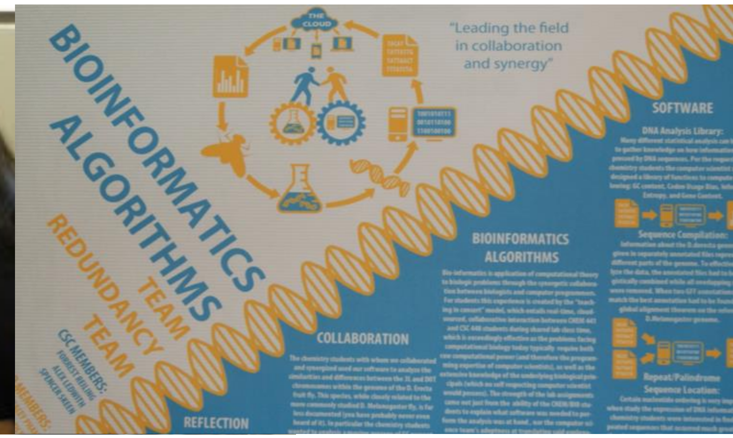
Statistical Software

Probability

Teaching *in concert*

a novel approach to interdisciplinary collaborative project-based instruction





Ed Himelblau

Biological Sciences

Anya Goodman

Chemistry and Biochemistry

Cal Poly

Learn by Doing

Alex Dekhtyar

Computer Science

Theresa Migler

Computer Science

In concert approach to the upper level bioinformatics course:

Lectures – separate, discipline-specific

BIO Applications course

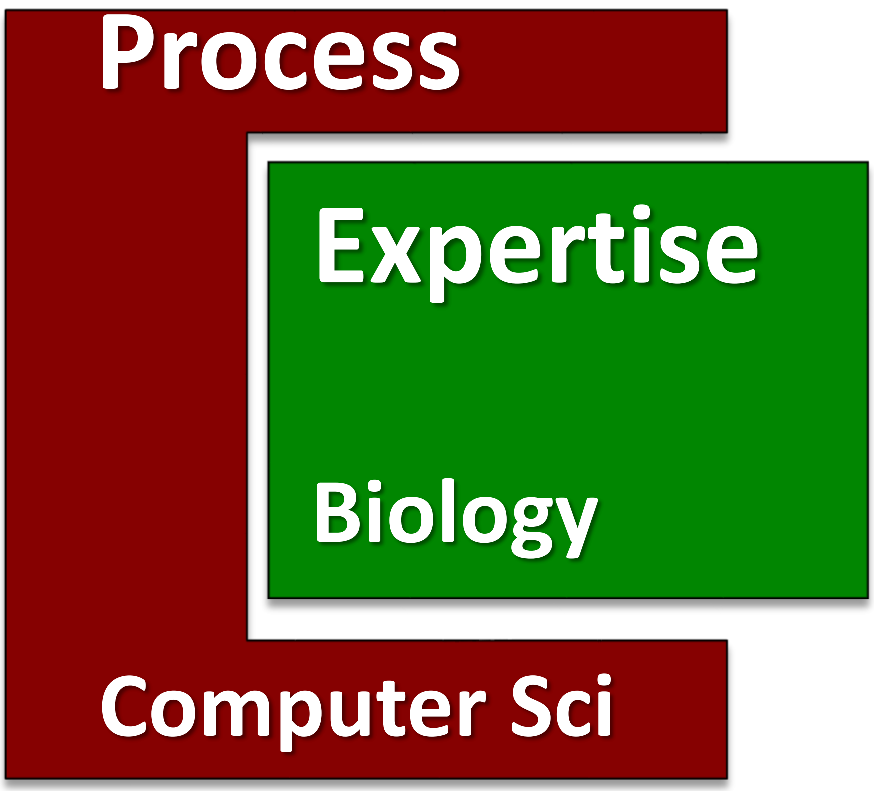
- Database and software use
- Genome research
- New technologies

CS Algorithms course

- Algorithms
- Data management
- Software Development

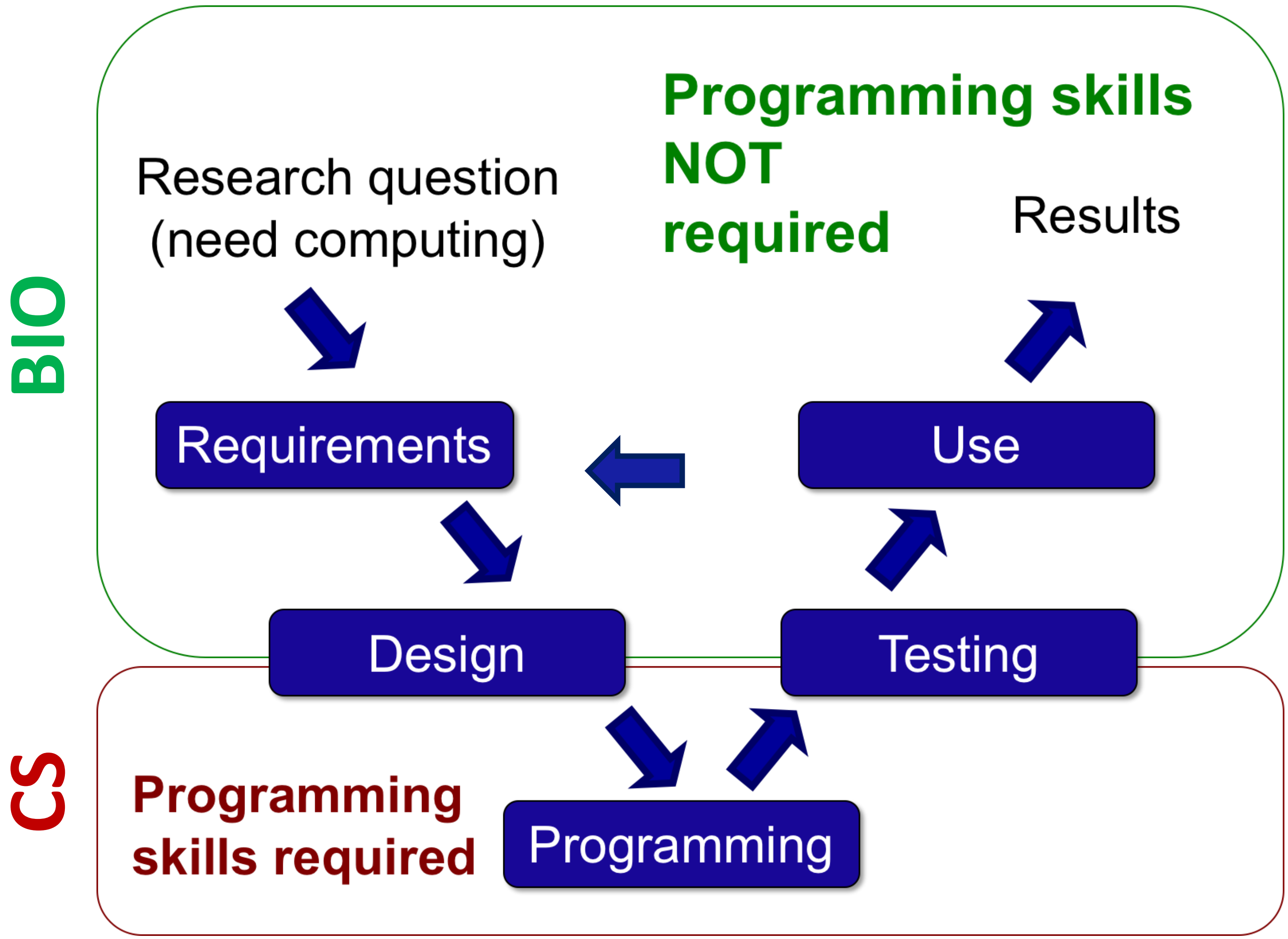
In concert approach to the upper level bioinformatics course:

Lectures – separate, discipline-specific



Laboratories – collaborative assignments structured around **clearly defined interdependent** roles

Interdisciplinary collaboration structured around **clearly defined interdependent** roles



Software development cycle

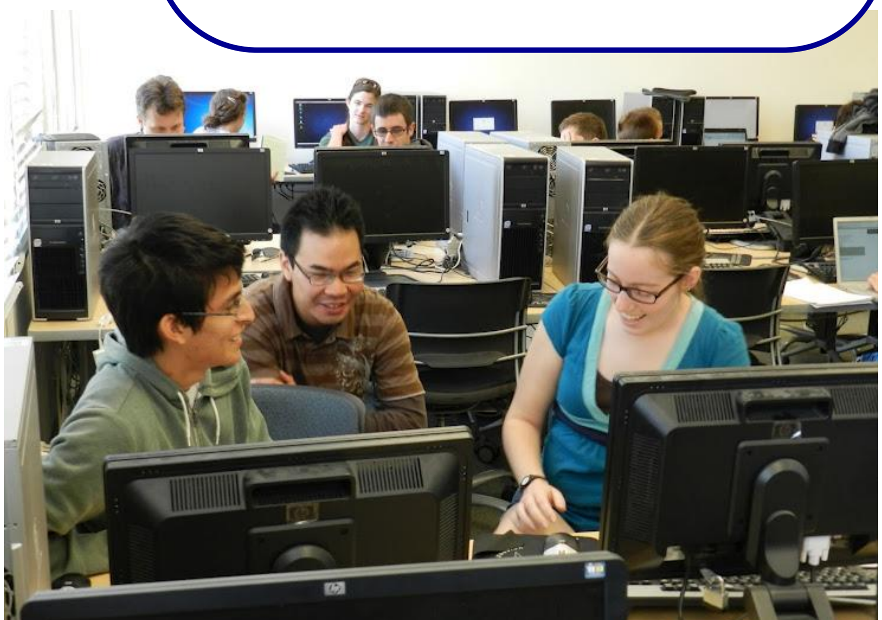
Interdisciplinary collaboration structured around **clearly defined interdependent** roles

BIO Applications course

- Discuss the problem
- Write program requirements
- Test software
- Use software

CS Algorithms course

- Elicit program requirements
- Design/build software
- Test software
- Deliver/maintain



In concert approach to the upper level bioinformatics course:

BIO Applications course

- Database and software use
- Genome research
- New technologies

Add:

- Comp. thinking

CS Algorithms course

- Algorithms
- Data management
- Software Development

Add:

- Effective collaboration across disciplines

In concert approach: implementation

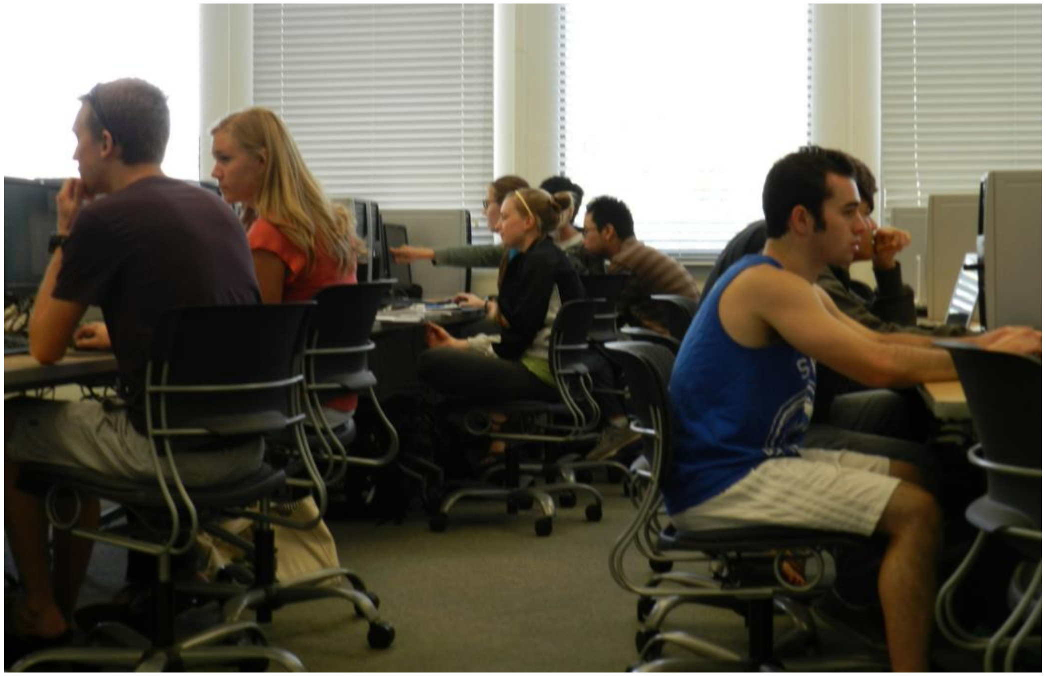
9-12 teams

30-36 CS students **2-3 CS**

20-25 BIO students **2-3 BIO**

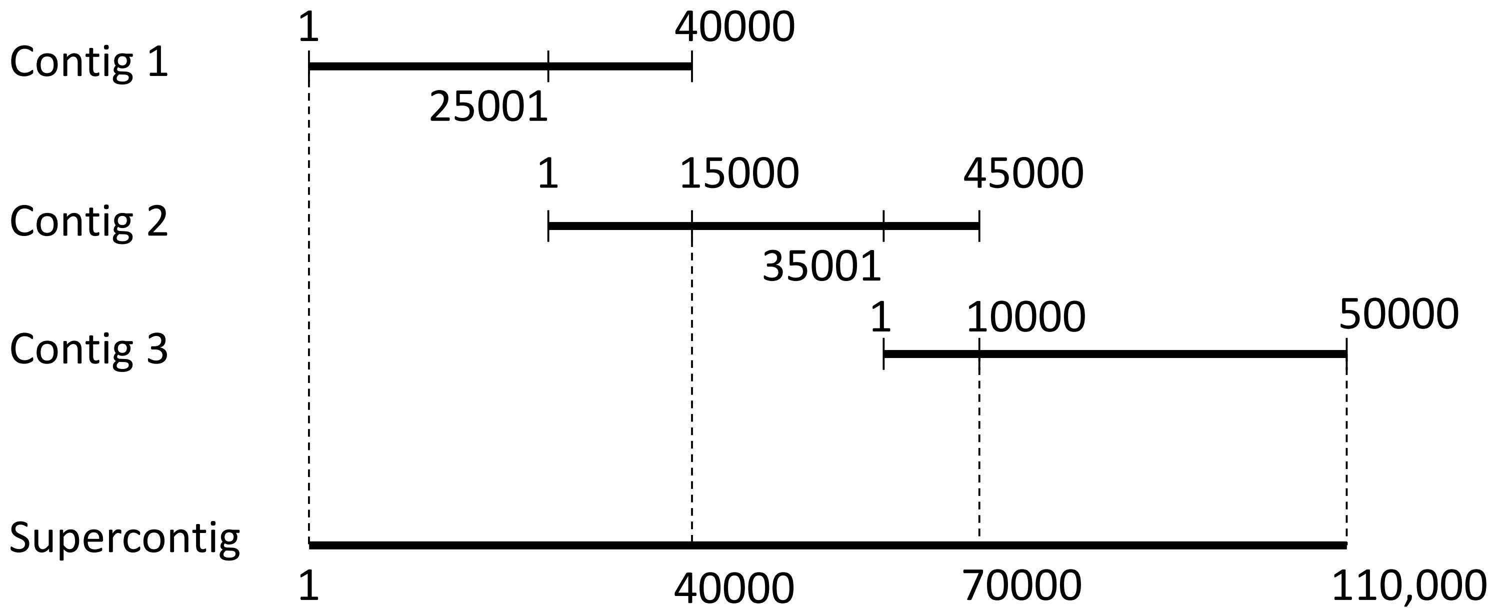
1-3 teaching assistants, two instructors

10-week quarter
60 hours
meet twice a week:
1.5 hr lec (separate)
1.5 hr lab (joint)



Example lab assignments:

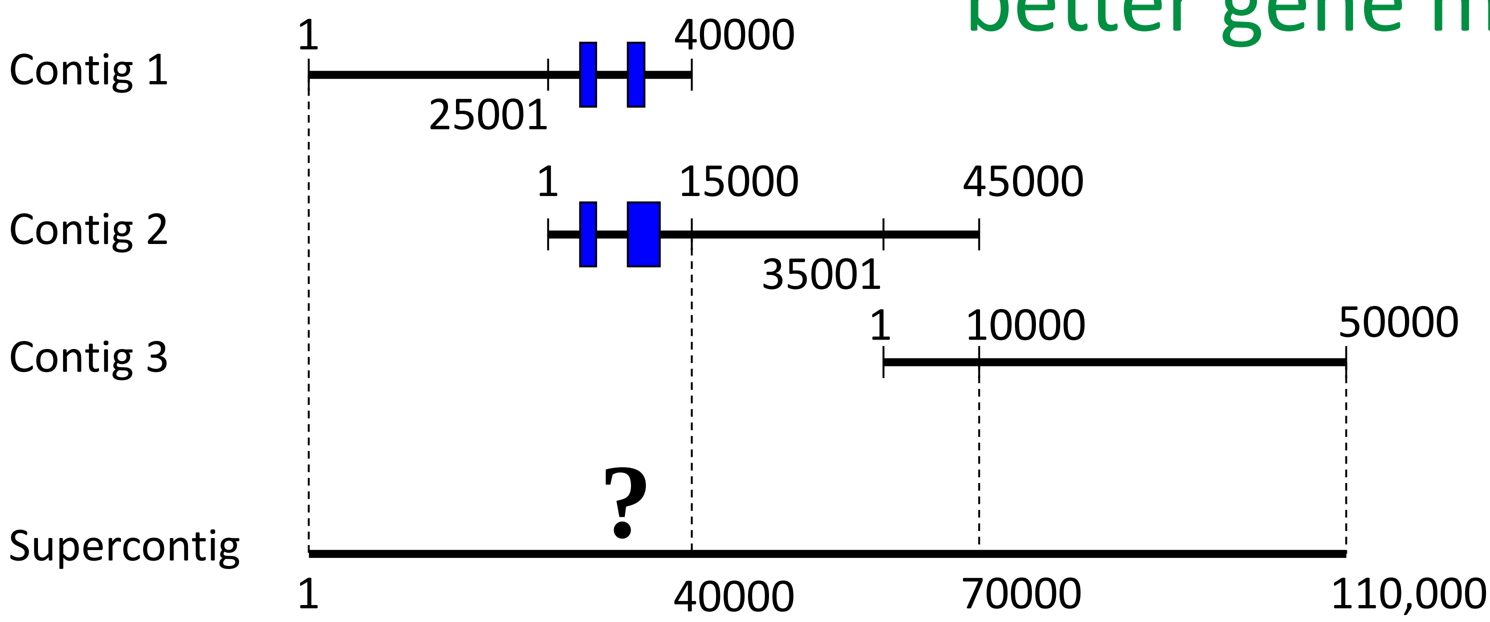
Question/Problem for BIO: How large is the genome region spanned by all annotated contigs?



Programs/algorithms for CS: Contig and GFF merger (string matching, data manipulation)

Example lab assignments:

Questions for BIO: For independently annotated genes, do annotations match? If not, which is a better gene model?



Programs/algorithms for CS: Annotation QC (global alignment)

***In concert* learning objective:** cooperate effectively with colleagues within and across disciplines

Assessment: student products and exit survey

Questions:

1. What were the **benefits** of working with partners from a different discipline?
2. What were the **challenges** of working with partners from a different discipline?
3. If you overcame the challenges, **how**?

1. What were the <u>benefits</u> of working with partners from a different discipline?

CS: "I liked that most of the pressure from the course came from our biology partners and having things ready for them, rather than pressure just to get a good grade."

1. What were the **benefits** of working with partners from a different discipline?

BIO: "This was a new experience … It required all to be respectful of each other's lack of understanding with the other's background but be able to clearly discuss each for precision + accuracy"

BIO: "Learning how different we think from each other and how to communicate more effectively."

2. What were the <u>challenges</u> of working with partners from a different discipline?

"Really the same as the benefits…

differences in background knowledge

and

communication."

2. What were the <u>challenges</u> of working with partners from a different discipline?

BIO: “They had little/no prior knowledge of the concepts, so everything had to be very clear and precise while also very detailed.”

CS: "Transferring expert knowledge between two disciplines is really hard.”

3. If you overcame the challenges, how?

"Dogged determination, sacrifice, and communication efforts/skills got us through.

When these failed, we failed."

"Learning to speak up when something is not correct on either side of the team and trying to teach each other bits of background info …"

3. If you overcame the challenges, how?

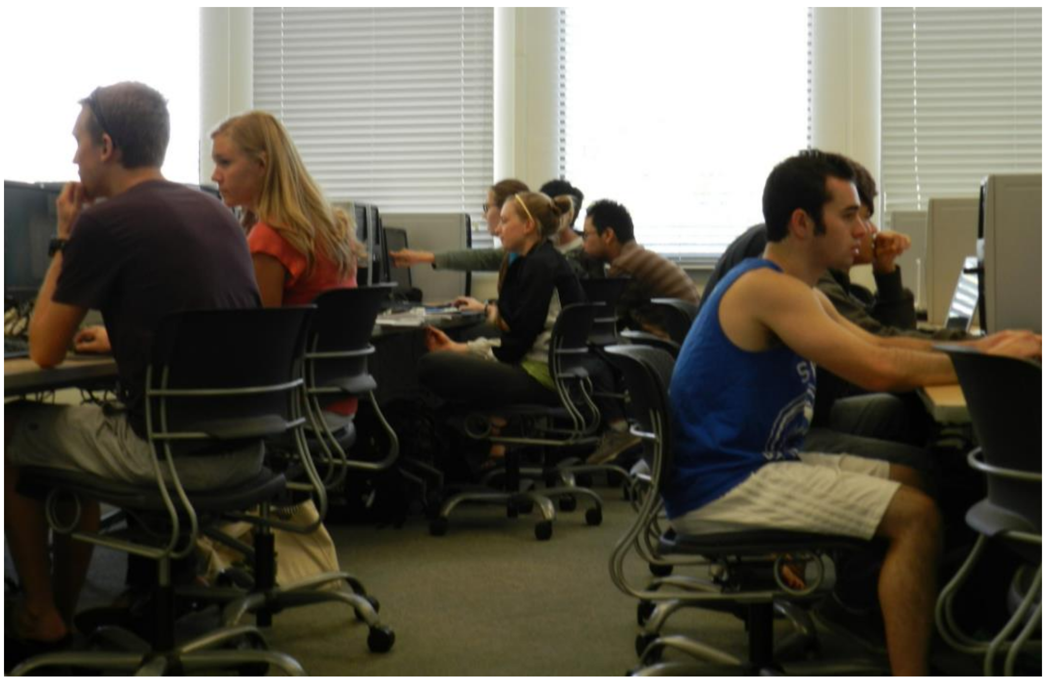
“by trying to fully understand the information before communicating it to CS partners”

“we had to discuss frequently and rewind our explanations until they made sense”

"We overcame the challenges by talking about the problem and teaching each other things that the other majors did not know"

Student challenges: bridging a cultural divide

Count: 1,2,3,…	Count 0,1,2,…
DNA Sequence	String
Do it over and over	Loop



Instructor challenges:

- Managing student researchers (keeping track of student progress, providing timely feedback).
- Institutional support (scheduling, rooms, release time for development assignments).
- Timing/organization of assignments.
- Assessment (computational thinking communication skills)

Acknowledgment:

Genomics Education Partnership

Sponsored by Washington University in St. Louis, HHMI and NSF

Cal Poly

Learn by Doing