In Zusammenarbeit mit David Eichler Class Diagram Nut Moveable Vector + position: Vector + size: number CanvasRenderingContext + velocity: Vector + x: number isEaten: boolean + y: number constructor (_position: Vector) constructor (_posittion: Vector) constructor (_size: number) + draw(_type: number): void + draw(): void + set(_x: number, _y: number + move(_timeslice: number): void + scale(_factor: number) + add(_addend: Vector) Squirrel Leaf Cloud + size: number - type: number - type: number + hunger: boolean + triggered: boolean + size: number + size: number jumpVelocity: Vector constructor (_size: number) + draw(_type: number): void constructor (_size: number) constructor (_size: number) + draw(_type: number): void + draw(_type: number): void + move(_timeslice: number): void Nut - Activity Diagram Moveable - Activity Diagra _position?: Vector _position: Vector _position: Vector _size: Vector constructor draw to 0, 0 save transform set position to _eventClientX, eventClientY set velocity to andom directio and length translate to position set_isEaten to false subtract canvas dimension from component add canvas dimension to component draw elipses draw V restore Cloud - Activity Diagram _position: Vector _size: Vector _size: number constructor draw nParticles: number = 20 radiusParticle: number = 20 position to 0, 0 particle = path with full circle with radius particle gradient. Radial with a = 0.5 -> a=0 set velocity to random direction Squirrel- Activity Diagram and length setTransform _position: Vector _size: Vector size: number constructor draw set size to _size save translate transform to 0.0 position translate to position set velocity to 0 [draw < nParticles] draw elipses x: number = (random - 0.5) * _size.x y: number = (- random) * _size.y set size to size rotate elipses set hunger to restore to x, y draw particle



